INSTRUCTION MANUAL

Volume 2

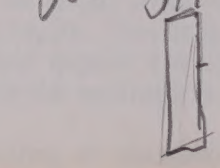
SD360 DIGITAL SIGNAL PROCESSOR

OPERATION

Preliminary Copy

(ALSO APPLICABLE TO SD360-75)

SPECTRAL DYNAMICS CORPORATION



INSTRUCTION MANUAL

Volume 2

SD360 DIGITAL SIGNAL PROCESSOR

OPERATION

Preliminary Copy

(ALSO APPLICABLE TO SD360-75)

August 1975

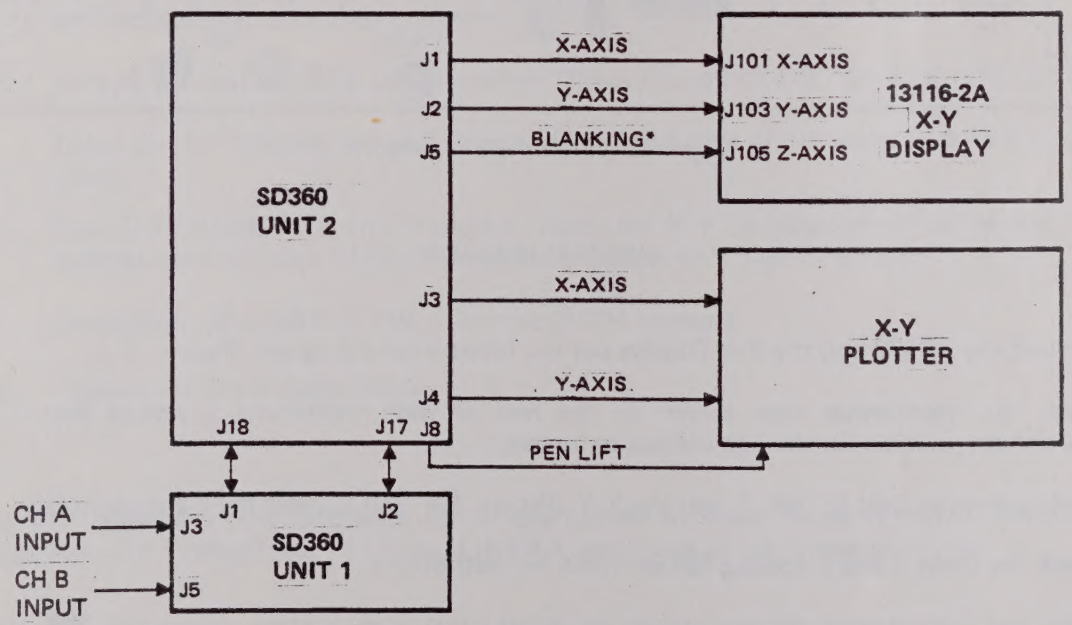
GENERAL

This section will familiarize the operator with the basic setups for each function. All possible setups will not be given; however, optional setups will be noted when applicable. The operator should read carefully all warnings and notes to obtain valid results. Particular applications for each function will follow the basic discussion. Some applications will require the use of more than one function. These applications will follow the last function needed for the application.

The material is arranged so that after completing all the basic system function setups, the operator should be familiar with all front panel controls and their interactions. He should be able to setup, calibrate and process data for an application of each function, choosing the proper methods of displaying and recording the data for optimum results.

INITIALIZATION

The Initialization procedure is to familiarize the operator with the SD360 front panel. This procedure includes calibration of the X-Y Display Oscilloscope. The setup is for a single channel signal input to be processed using the Function 2 1024 point Fourier transform. The X-Y Display will be the magnitude and phase of the CH A input signal. The magnitude will be Y1 axis with Log amplitude, and Y2 axis will be phase with Linear amplitude, and the X axis will be Linear frequency.



*Blanking from SD360 must be connected to the oscilloscope Z-Axis input, not the blanking input.

Figure 1. Basic Interconnect

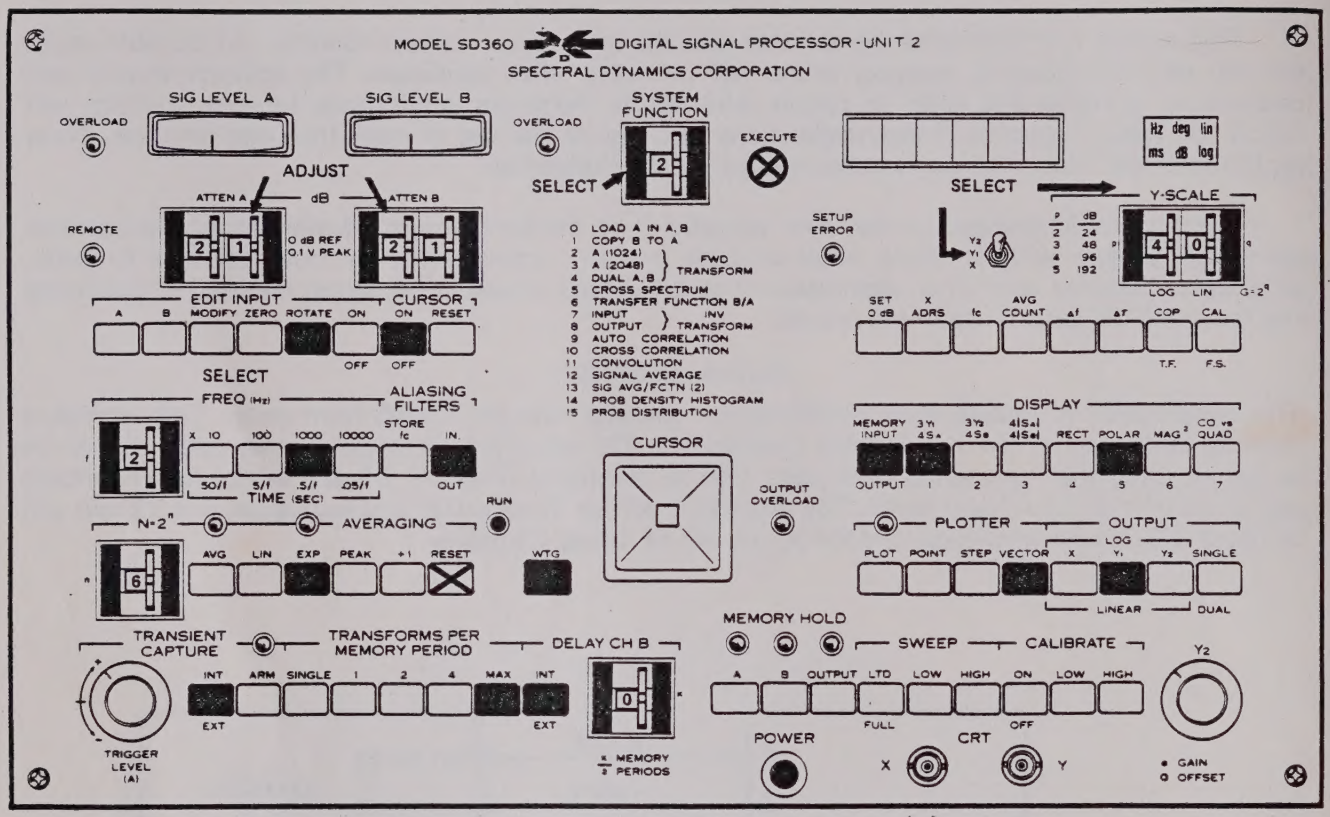


Figure 2. Initialization Diagram

INITIALIZATION

1. Connect the SD360 and the X-Y Display per the interconnect diagram, Figure 1.

2. Check the transformer slide switch on the rear of each instrument to ensure that it is in the appropriate position for the line voltage to be used.

3. Apply power to Unit 1, Unit 2, and the X-Y Display. On Unit 2, reset the p switch to 0.*

4. Check the Units 1 and 2 cooling fan air inlets for restrictions.

5. Using the Initialization diagram, select the front-panel push buttons designated ■ and set the thumbwheel settings to number X designated by [X]. The MEMORY HOLD lights should be off. Momentarily press the controls designated ⊠ or ⊗. Lights designated ◉ should be lighted.

6. Apply a sinusoidal input to Unit 1 CHA and CH B within the selected Frequency Range and the ATTEN A and B settings. Signal input to CH B is for viewing CH B input memory during calibration.

*Whenever power to Unit 2 has been off, reset the p switch to 0 before setting to the position desired for LOG display.

7. On the X-Y Display, adjust the X and Y axis GAIN and OFFSET controls until the CH A signal input (Y1) can be observed. The controls to adjust the Y2 Gain and Offset are on UNIT 2 front panel. The vertical position of the CH B signal input is controlled by the Y2 OFFSET control (outer rotary dial) and the Y2 GAIN control is the inner rotary dial.

8. Calibrate the X-Y Display as follows:

   a. Push the CALIBRATE ON, LOW and HIGH.

   b. Release the MEMORY INPUT control. The display will now be Y2 Cal signal output. The horizontal line of the Y1 display represents:

      –80 dB for a p switch setting of 4.
      –40 dB for a p switch setting of 3.
      –20 dB for a p switch setting of 2.

   c. Release the CALIBRATE LOW control. The horizontal line shown on the Y1 display represents the 0 dB Cal level.

   d. Push CALIBRATE LOW.

   e. Alternate between steps 8c. and 8d. while adjusting the X-Y Display GAIN and OFFSET controls for a LOG amplitude display of Y1 data.

   f. Select the Y2 position on the toggle switch.

   g. On Unit 2 adjust the Y2 GAIN and OFFSET controls while alternating between steps 8c. and 8d. for LINEAR amplitude display of Y2 data.

   h. Release the CALIBRATE HIGH control. Check to see that CALIBRATE LOW is in.

   i. Using the X-Y Display control, adjust the lefthand edge of the display for zero frequency start point.

   j. Push the CALIBRATE HIGH control. Using the X-Y Display control, adjust the X-axis GAIN control for a full scale LINEAR Frequency display of Y1 and Y2 data.

   k. Release the CALIBRATE ON, LOW and HIGH controls.

   l. Disconnect CH B signal input.

The X-Y Display is now calibrated for a data display of linear X-axis (LINEAR FREQUENCY), LOG amplitude of Y1 (Magnitude of CH A), LINEAR amplitude of Y2 (Phase of CH A).

NOTE

To view X, Y1, or Y2 axis data with LOG scaling, push in the appropriate OUTPUT button. To view any axis in LINEAR scaling, release the appropriate OUTPUT button.

FUNCTION 0 – LOAD A IN A, B

Function 0 may be used whenever processing of data by more than one method is desired. It is particularly useful in processing transient data. Used in conjunction with the TRANSIENT CAPTURE mode, Function 0 places the CH A transient input data into CH A AND CH B input memories simultaneously and automatically places the memories in HOLD. Data in memory A may be processed or edited in a variety of ways. If the data in Memory A is altered or destroyed in processing, the original data stored in Memory B may be recovered. See Function 1. Figure 3 is the data flow diagram. Figure 4 is a typical Function 0 plot of the contents of the input memories.

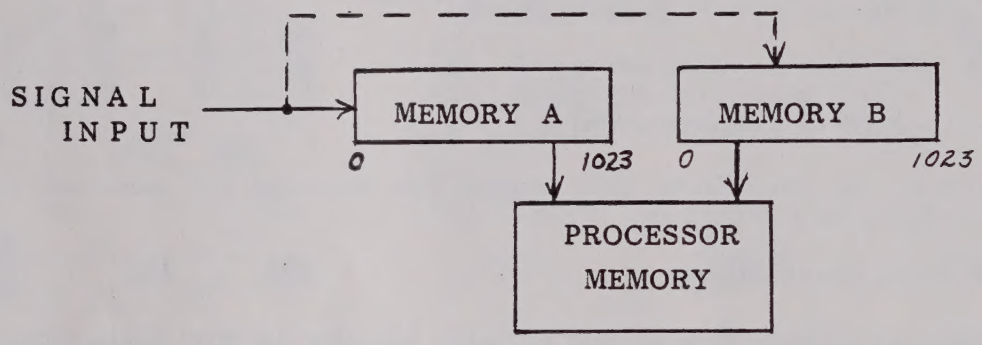
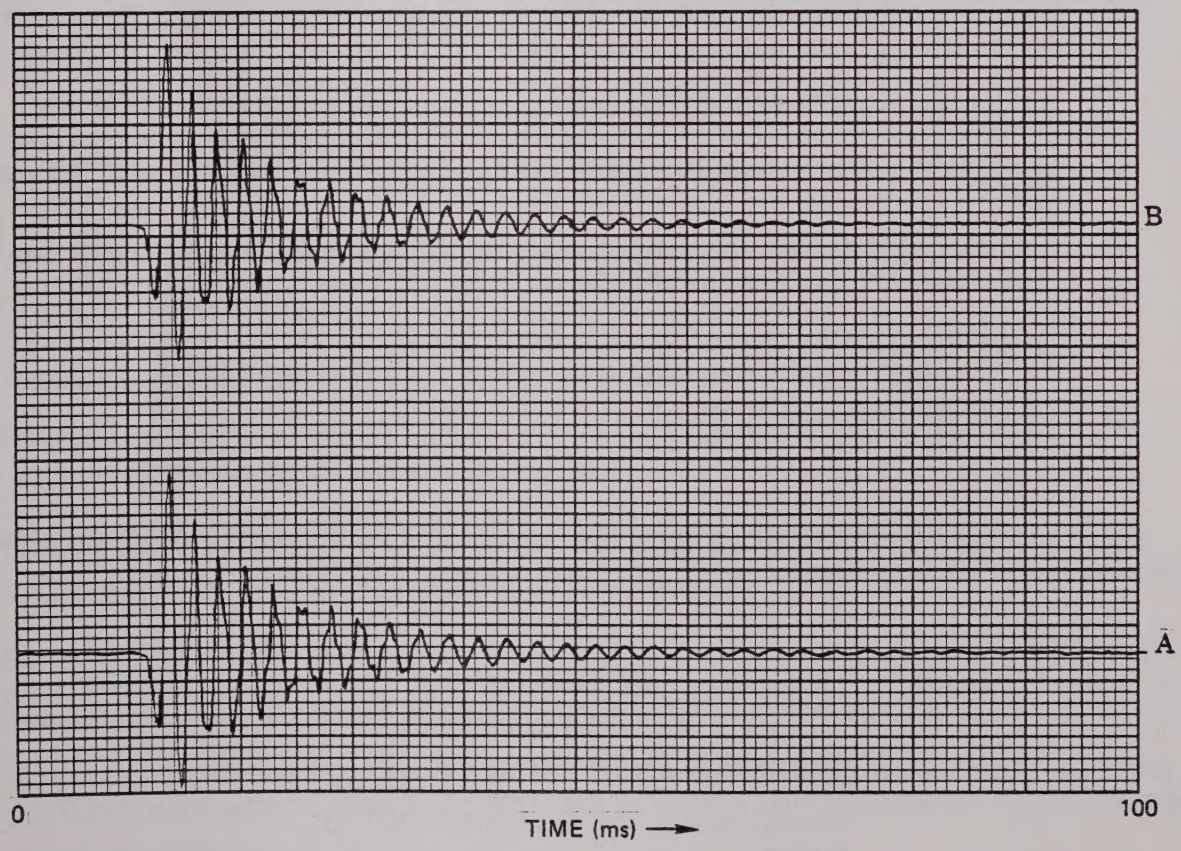


Figure 3. Data Flow For Function 0

Figure 4. Typical Function 0 Plot



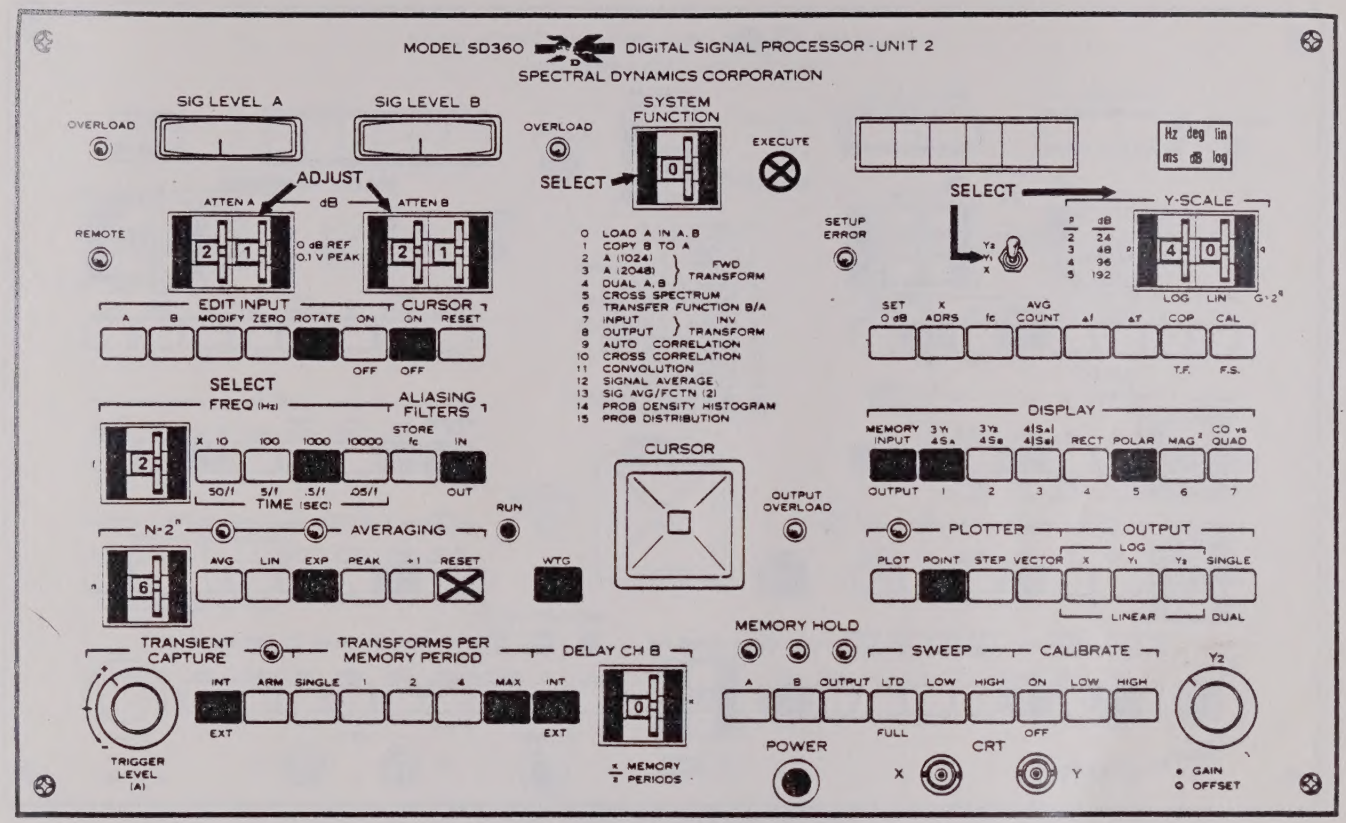


Figure 5. Function 0 Diagram

FUNCTION 0 — Load A in A, B

1. Using the Function 0 diagram, select the front-panel push buttons designated ■, and set the thumbwheel switches to X as designated by [X].
2. Apply a continuous data signal to CH A within the Frequency Range and ATTEN A settings.
3. Momentarily push the AVERAGING RESET control, ⊠.
4. Push the EXECUTE button and data will be loaded into CH A and B input memories, ⊗.
5. Push the MEMORY HOLD A button, which simultaneously holds both CH A and B.

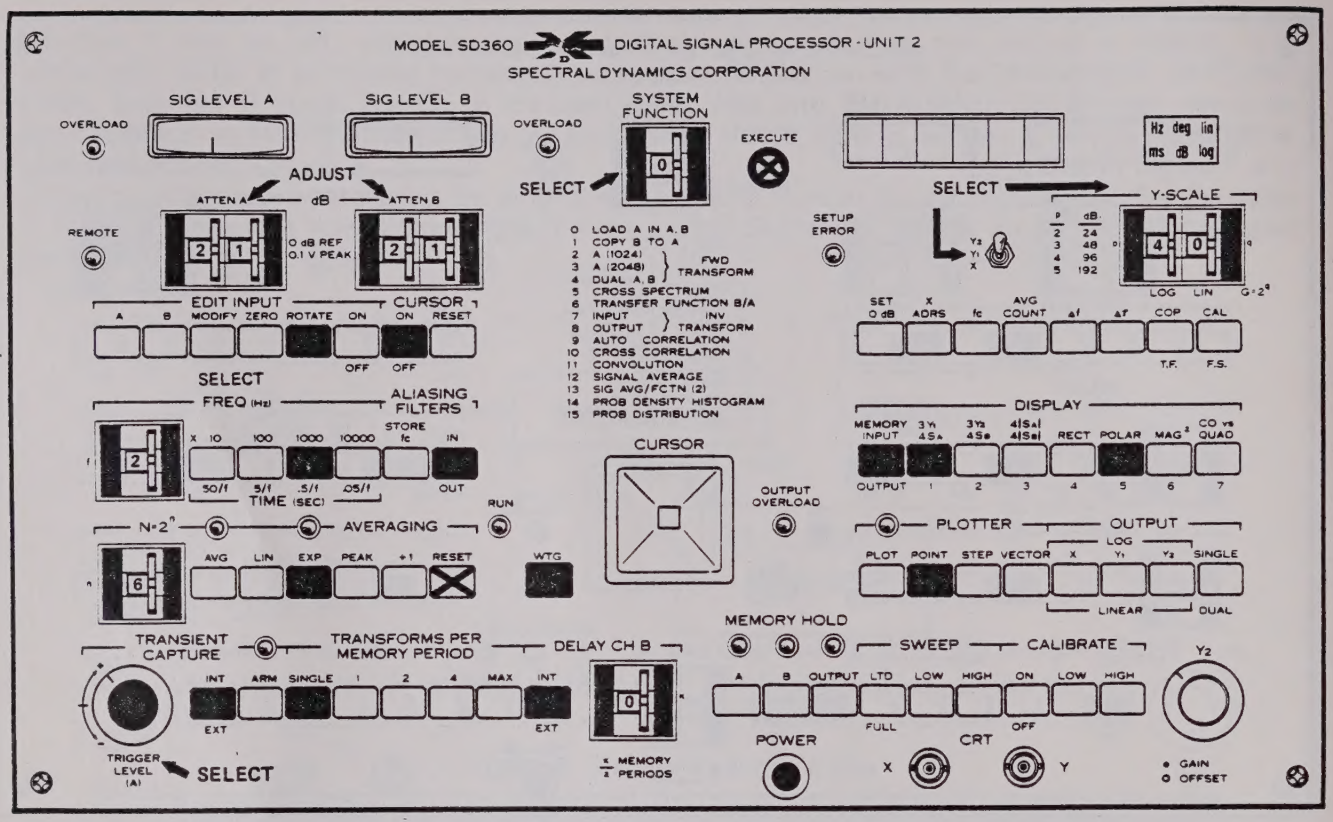


Figure 6. Function 0 — Single Transient Capture

A single transient can be loaded into both input memories with Function 0 using the TRANSIENT CAPTURE feature. In order to observe the Trigger Level, before performing the following setup, set the function switch to Function 2 and press EXECUTE to ascertain that the input memories are clear.

1. Using Single Transient Capture diagram, select the front-panel pushbuttons designated, and set the thumbwheel switches to the number X as designated by [X].

2. Select an appropriate data Trigger Level (A) setting by pulling out the rotary control and observing the threshold level on the X-Y Display, in Y2 memory location. The amplitude of the threshold level can be read out on the LED display in terms of linear counts with 2047 the maximum count. The ratio of actual count:2047 can be used as an indication of the percentage of the selected Input Voltage range that the threshold represents.

3. Push in TRIGGER LEVEL (A), being careful not to change the setting.

4. Push TRANSIENT CAPTURE ARM push button. TRANSIENT CAPTURE ARM light should come on.

5. Momentarily push the AVERAGING RESET and EXECUTE push buttons. MEMORY HOLD A and B lights should be off.

6. Apply the transient signal to CH A. MEMORY HOLD A and B lights will come on when the transient is captured and TRANSIENT CAPTURE ARM light will go off. The input transient (CH A) has now been captured in CH A and CH B input memories.

7. Adjust the Y2 GAIN and OFFSET controls, if necessary, to view CH B data.

THIS PAGE INTENTIONALLY
LEFT BLANK

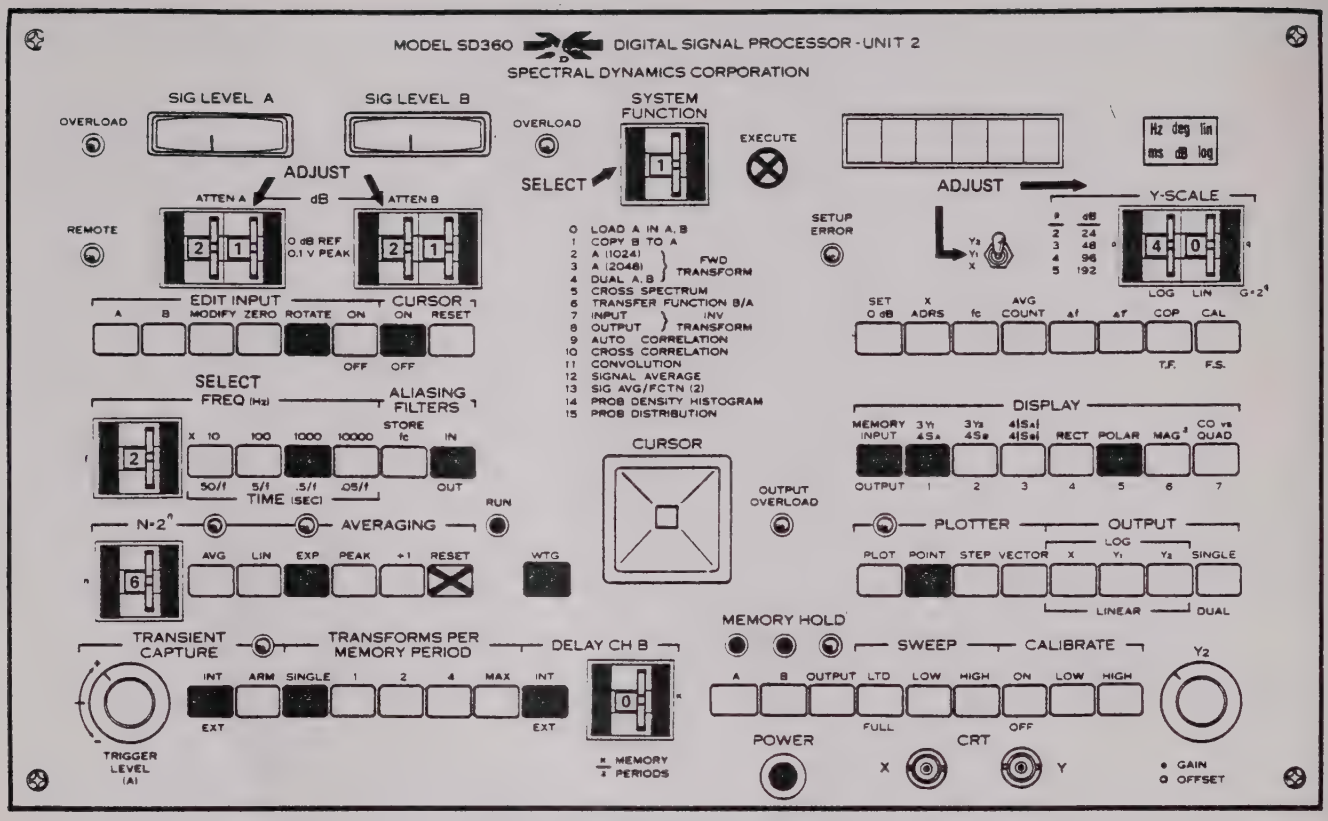


Figure 10. Function 1 Diagram

FUNCTION 1 – COPY B to A

A copy of the data in Memory B, usually a captured transient, is placed into Memory A, leaving the contents of Memory B intact.

Function 1 assumes that there is valid data stored in MEMORY B, usually the result of performing a Function 0 at some earlier time.

If a Function 0 has been performed and the CH B data held in memory, any number of Function 1 operations may be performed followed by any other chosen processing function on CH A data.

1. The MEMORY HOLD A and B lights should be lighted.

2. Using the Function 1 diagram, select the front-panel push buttons designated ■. Momentarily push the control marked ☒. Push EXECUTE ⊗.

3. Release MEMORY HOLD A and push EXECUTE until all the data has been transferred to CH A from CH B. It may be necessary to push EXECUTE more than once to achieve a complete data transfer.

4. Push MEMORY HOLD A. Data may appear to jump due to the fact that pushing MEMORY HOLD push button will cause rotated data to be placed in the proper location. MEMORY HOLD A and B will light. Proceed with further editing and processing of CH A data.

FUNCTION 2 – FORWARD TRANSFORM A (1024)

Function 2 generates 512 frequency domain coefficients, in either rectangular or polar coordinates, from 1024 time domain samples from Channel A signal input. These data samples (words) are transferred from input memory A into the processor, multiplied by the selected weighting function and placed into processor memory. A Fourier Transform of the data is performed, the selected coordinate conversion is made and the resulting data placed in output memory for display. Figure 11 gives the data flow for Function 2.

$S_x(j\omega)$ = Forward Transform

$S_x(j\omega) = a_n + jb_n$

$$a_n + jb_n = \frac{1}{N} \sum_{\kappa=0}^{N-1} Y(\kappa) W^{n\kappa}$$

where $W = e^{-j\frac{2\pi}{N}}$, smallest resolvable increment on the perimeter of the unit circle.

$n = 0, 1, 2, \ldots\ldots, \frac{N}{2} - 1$

n = Spectrum Component Number

$\kappa = 0, 1, 2, \ldots\ldots, N-1$

κ = Sample Number

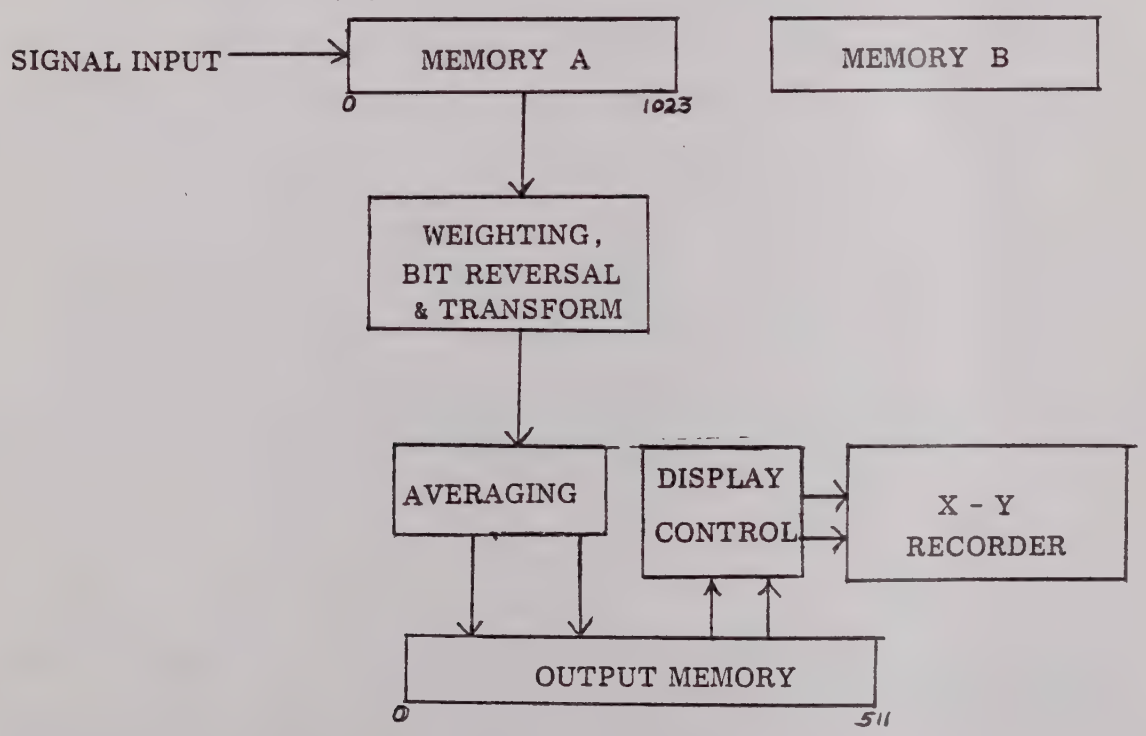


Figure 11. Data Flow For Function 2

FUNCTION 2 – CH A 1024 POINT FOURIER TRANSFORM

1. Using the Function 2 Diagram, select the front-panel push buttons designated ■ and set the thumbwheel switches to the number X designated by ⊠.

2. Apply the data to CH A input and select the desired Frequency Range. Adjust the ATTEN A setting for maximum SIG LEVEL A without OVERLOAD.

3. Momentarily push the control marked ⊠. Push EXECUTE ⊗.

4. The X-Y display will be the unaveraged spectrum of the data signal displayed as LOG amplitude versus LINEAR frequency (Figures 13 and 14). By using the CURSOR control in conjunction with the 6-digit LED readout, a selected peak of interest may be precisely defined.

   a. Set the Cursor dot to the peak of interest.

   b. The LED readout gives the LOG amplitude of this peak in dB.

   c. Change the toggle switch from the Y1 axis to the X axis. The frequency of the peak of interest is shown in Hz. Return the toggle switch to the Y1 position.

   d. Push the Δf push button. The X-axis filter spacing is shown in Hz.

   e. Change the display to Linear amplitude by releasing the OUTPUT LOG Y1 push button. Adjust the Y-SCALE switch (q switch where the output Gain = 2^q).*

   f. Change the output display to rectangular coordinates† by selecting DISPLAY RECT, Y1 on the axis toggle switch, and releasing the OUTPUT SINGLE push button (Figure 15).

   g. Change the output display to linear squared amplitude and phase versus frequency by selecting DISPLAY MAG^2 and releasing OUTPUT Y1 and Y2 push button. Either LINEAR or LOG X may be selected (Figure 16).

---

*The q switch is in the calibration signal path and any change to it requires a comparable change in the X-Y recorder amplitude scaling.

†The Y2 Gain and Offset controls will usually require adjustment to get the display in the desired position for viewing.

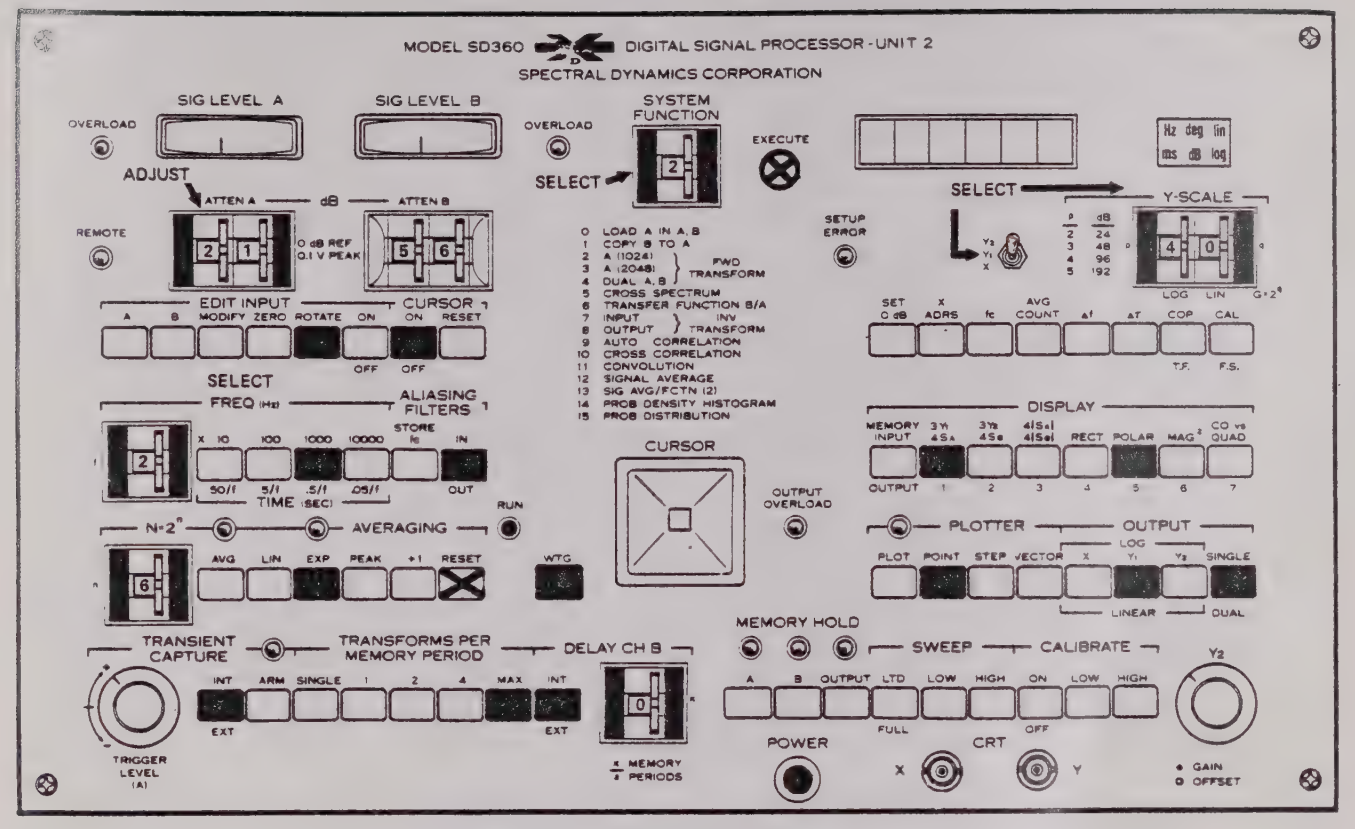


Figure 12. Function 2 Diagram

FUNCTION 2 – CH A 1024 POINT FOURIER TRANSFORM WITH AVERAGING

1. Use the Function 2 Diagram for the basic setup. The following controls may be changed to other settings if desired.

   a. Select number of averages desired on the n switch, $N = 2^n$.

   b. Select averaging mode: LIN, EXP or PEAK.

   c. Select TRANSFORMS PER MEMORY PERIOD: 1, 2, 4 or MAX; or, for statistically independent data averaging, select 1 TRANSFORM PER MEMORY PERIOD.

2. Push the AVG push button. The Averaging light will be on when selected number of averages have been accumulated in th LIN mode. If EXP or PEAK modes were selected, the Averaging light will flash on and off. Figure 17 is a typical Function 2 with averaging.

X-Y PLOTTER CALIBRATION

The following diagram is to calibrate the X-Y Plotter for Function 2 single channel signal input with LOG amplitude and LINEAR frequency.

1. Set up Unit 2 controls per Function 2 Diagram.

2. Connect plotter as shown in the interconnect diagram (Figure 1).

3. Energize the plotter servo.

4. Press CALIBRATE ON and LOW push buttons. Release CALIBRATE HIGH.

5. Adjust the plotter X-axis ZERO control for 0 Hz. Adjust the plotter Y-axis ZERO control for the –80 dB point which corresponds to the CAL level for a p switch setting of 4. The CAL level for a p switch setting of 3 is –40 dB; for a p switch setting of 2, it is –20 dB.

6. Release the CALIBRATE LOW and set the CALIBRATE HIGH push buttons.

7. Adjust the plotter X-axis GAIN control for the full-scale frequency displayed in the LED readout. Adjust the Y-axis Gain control for 0 dB point on the graph paper.

8. Repeat steps 4 through 7 until the plotter X and Y-Axis spans are properly adjusted. If SWEEP LIMIT has been selected, CALIBRATE HIGH and LOW frequencies will correspond to the UPPER and LOWER SWEEP LIMIT points.

CAUTION

When performing this Calibration sequence, the LOW push button should be released when the HIGH push button is selected, and vice versa. This is to ensure a valid frequency readout on the LED readout during calibration.

9. Release the CALIBRATE HIGH and LOW push buttons.

10. Reset the p switch by turning it to 0 and then to position 4.

11. Position the cursor dot on the top of the fundamental and set to 0 dB by pushing the SET 0 dB push button.

12. Adjust the plotter Y-axis ZERO control until the pen lines up with the 0 dB point on the graph.

13. Check for a zero offset by setting p switch to position 1 and observe the readout. Readout should be 0. If so, return p switch to 4; if not, see caution note below. Readout gives the number of counts when p switch is set to 1. The units are not shown on the LED.

14. Release the CALIBRATE ON push button.

CAUTION

Vertical movement of the Cursor control will roll the data up or down on the display, thus affecting the 0 dB position of the plotter pen. To check if there has been an offset introduced, set the p switch to position 1 and observe the readout. The reading should be 0. If there is offset present, set the p switch to 0 and then to position 4.

Figure 13. Function 2 — Polar Log

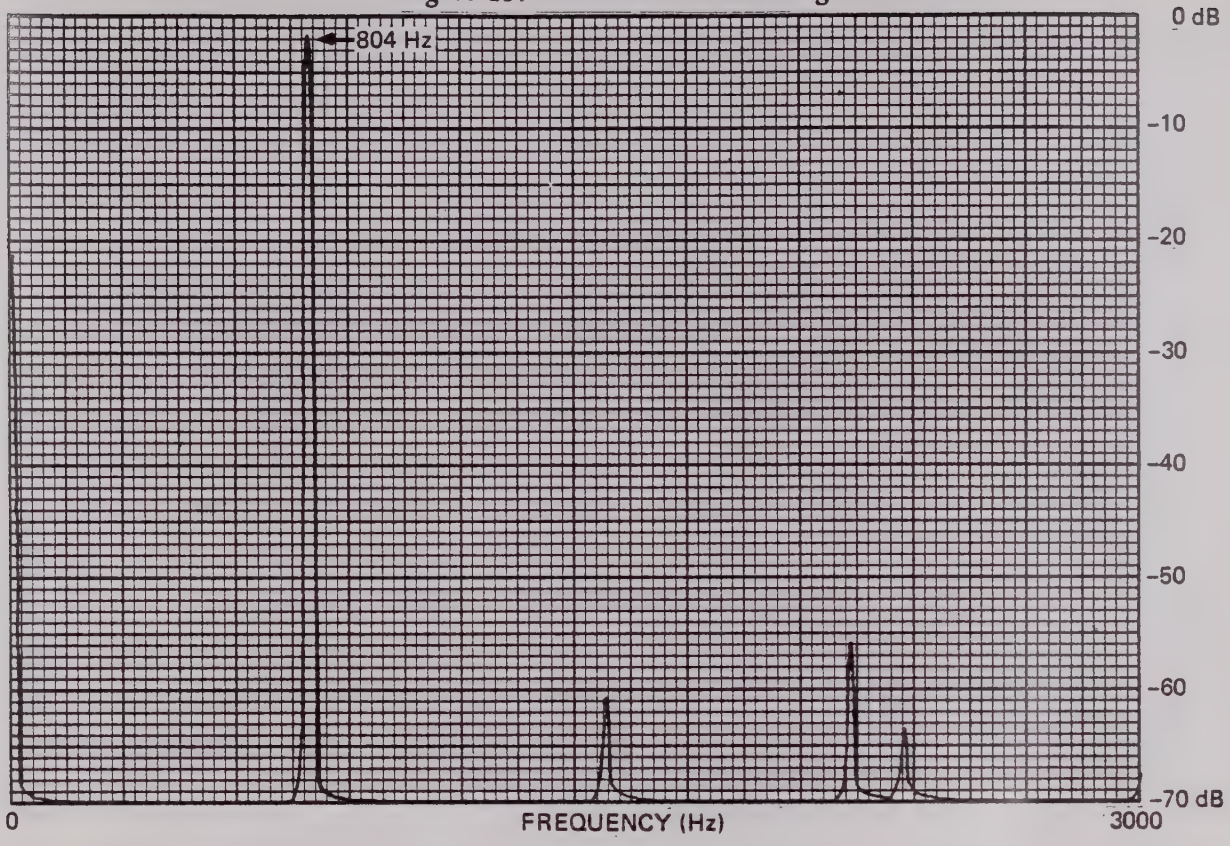


Figure 14. Function 2 — Polar Log

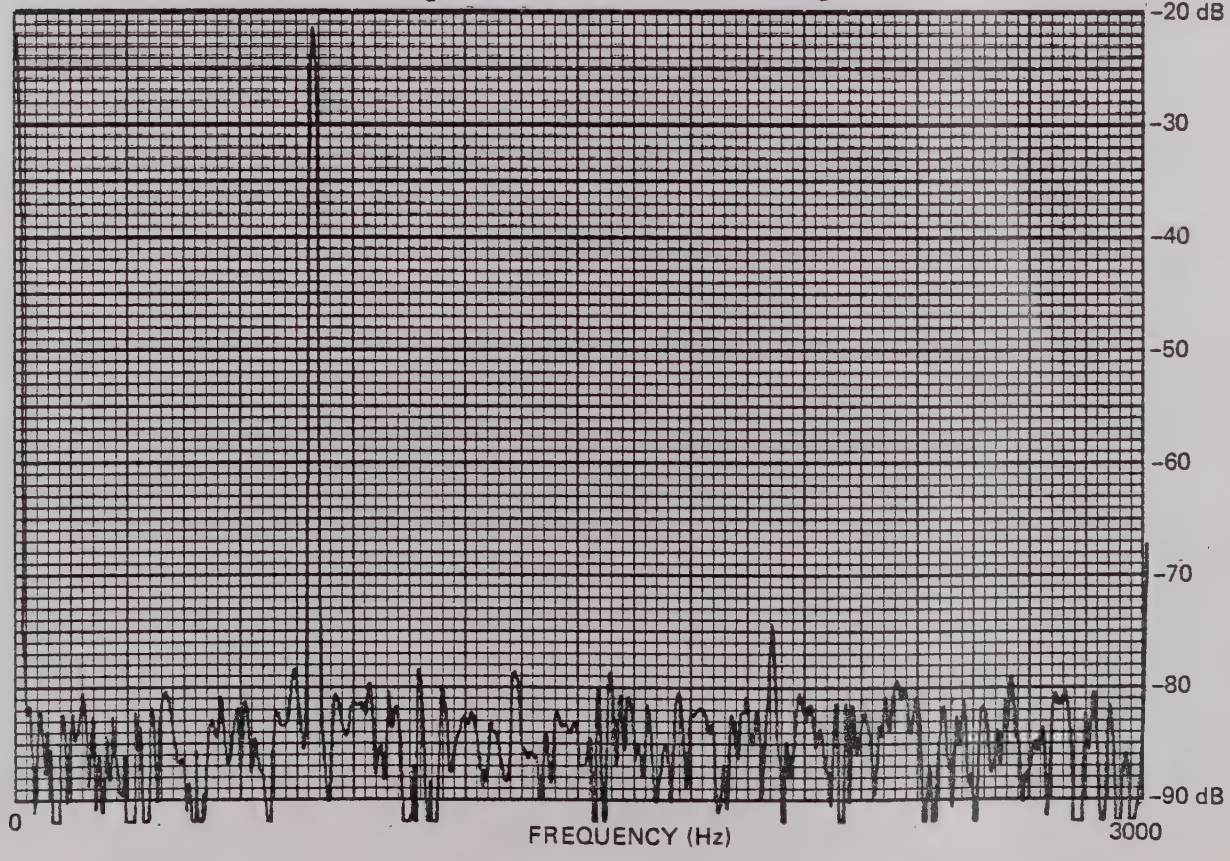


Figure 15. Function 2 — Rectangular

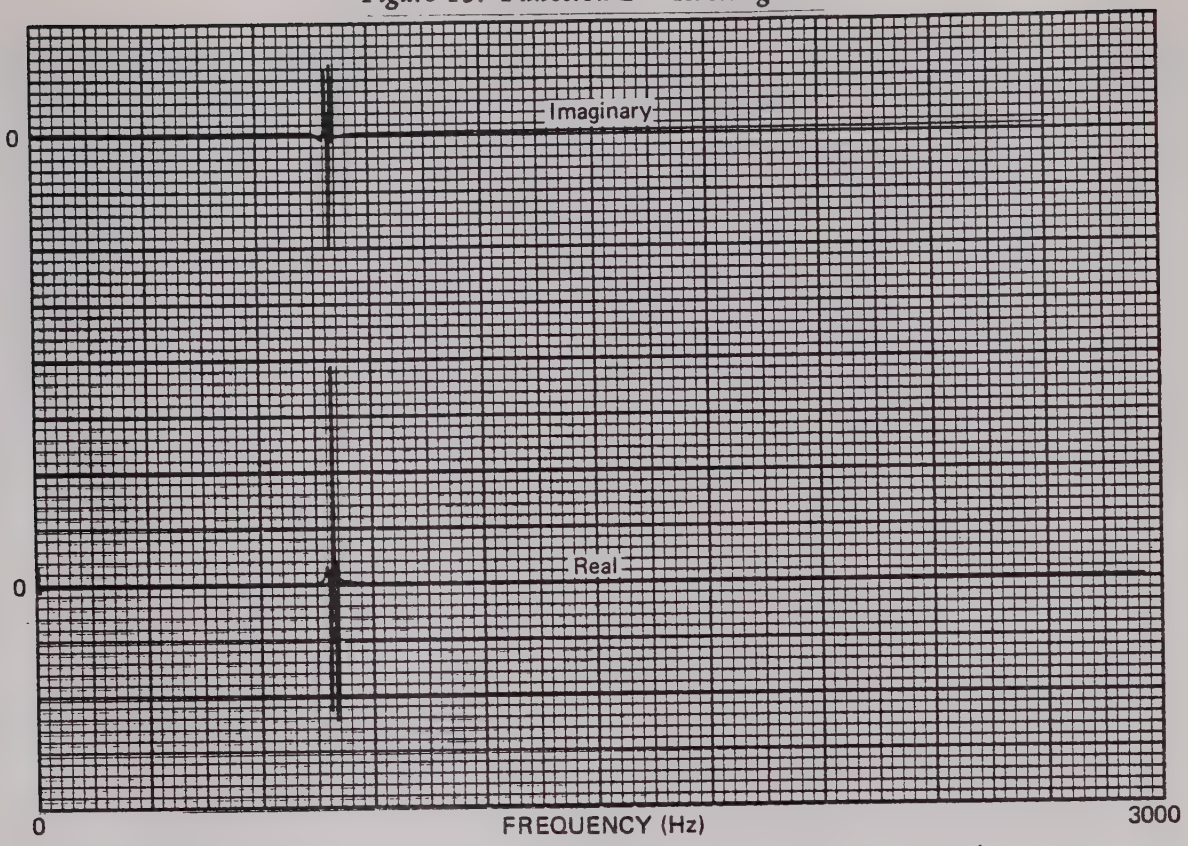


Figure 16. Function 2 — Mag^2 Linear

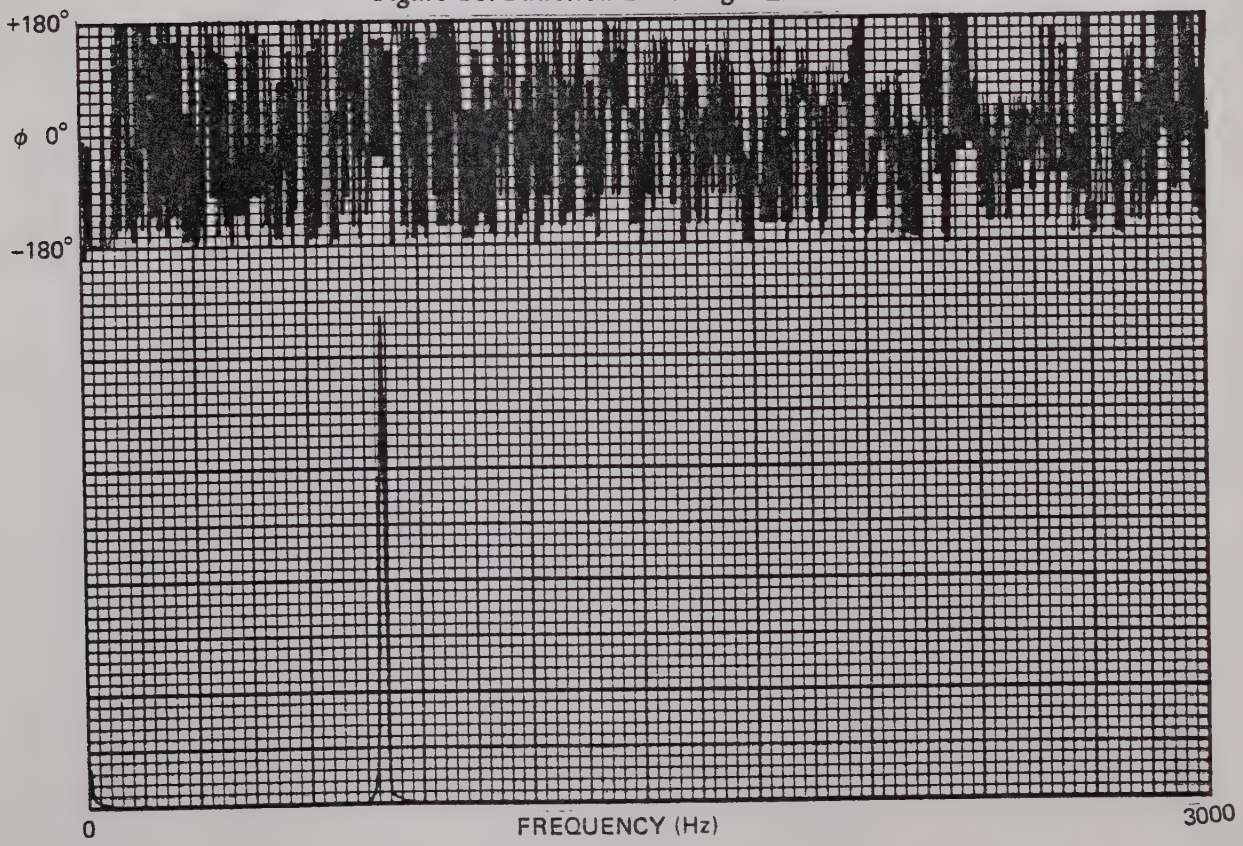


Figure 17. Function 2 — Polar Log Averaged

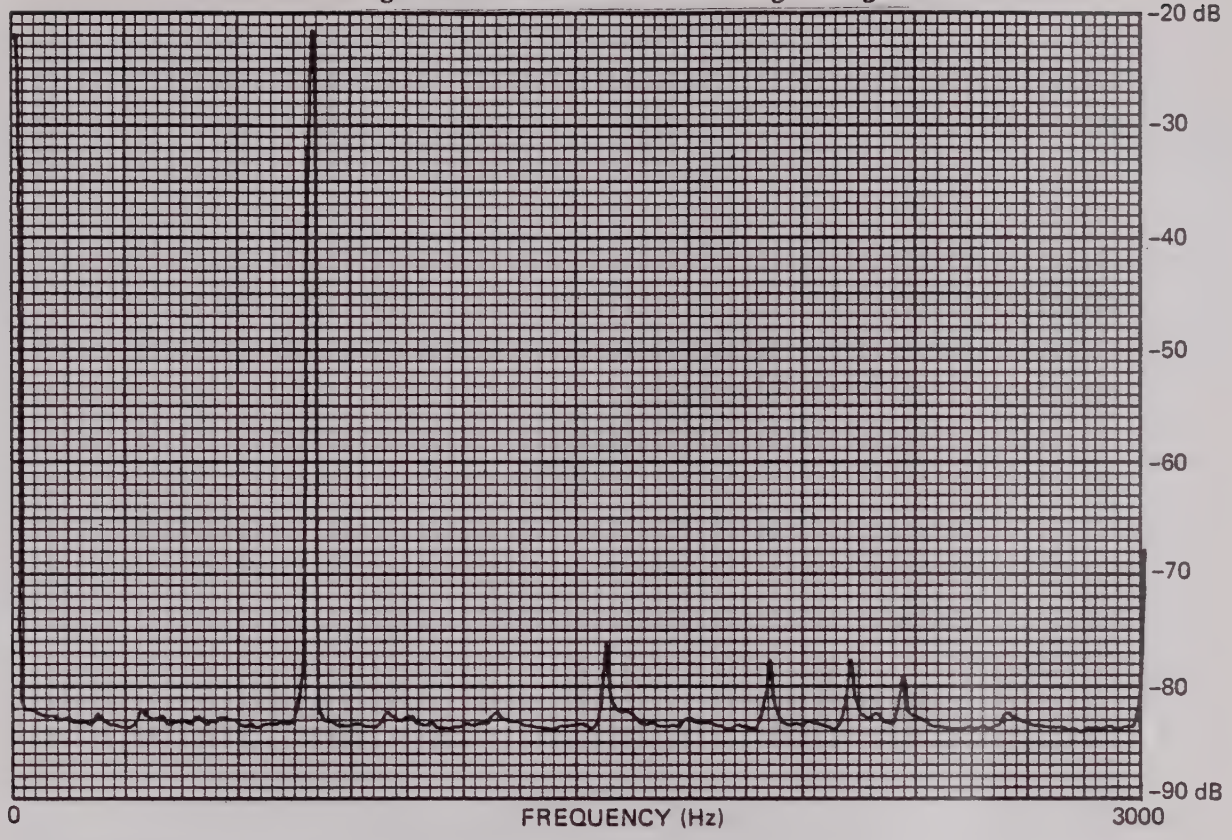
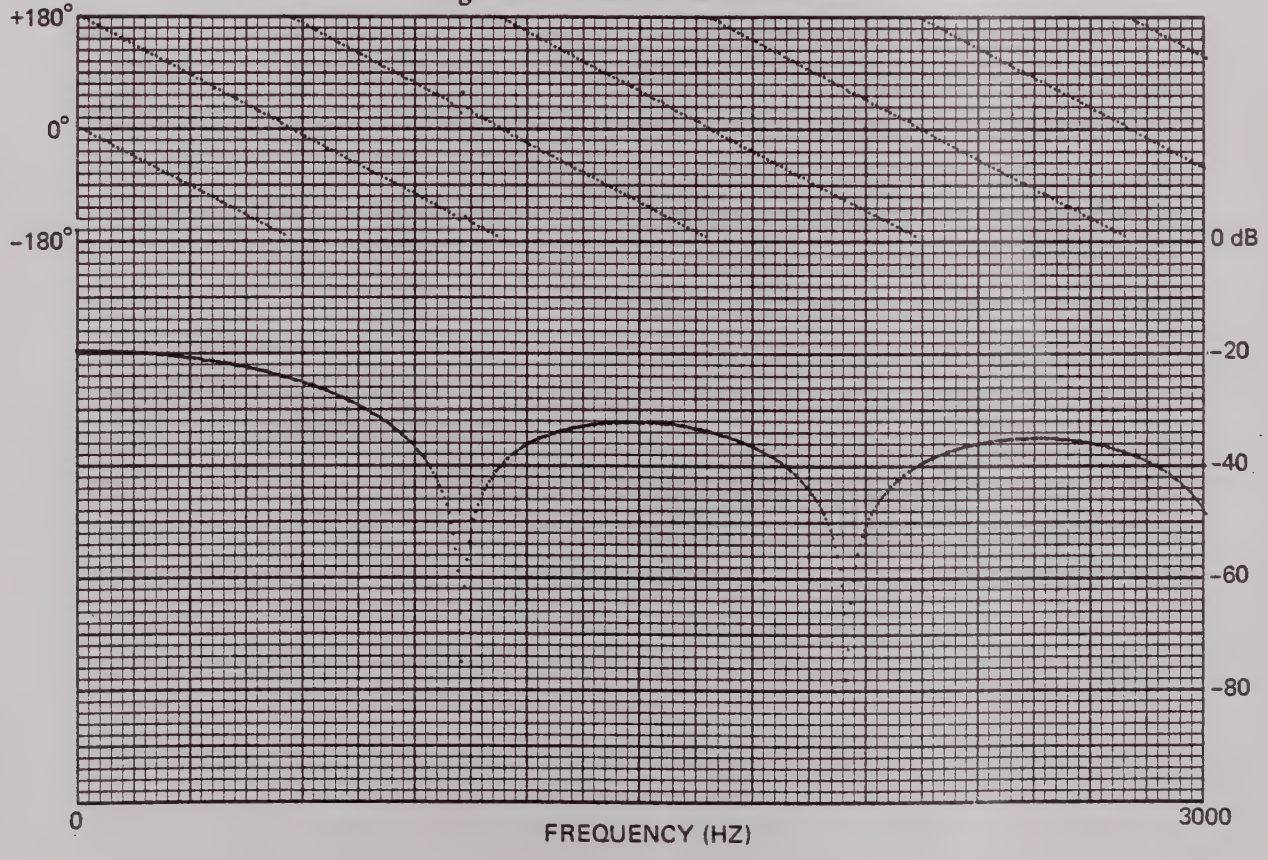


Figure 18. Function 2 — Polar



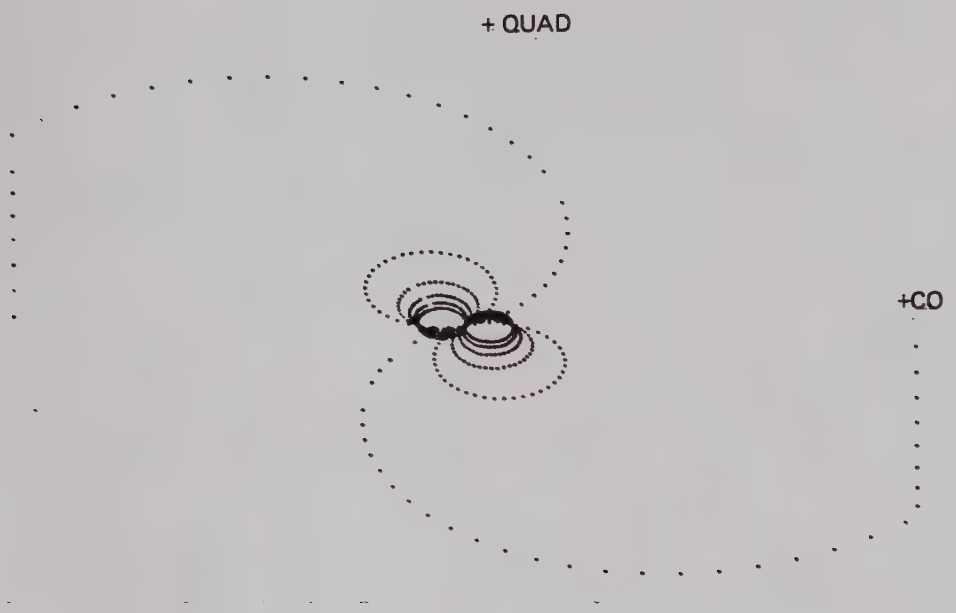


Figure 19. Function 2 – Nyquist Plot (Co vs Quad)

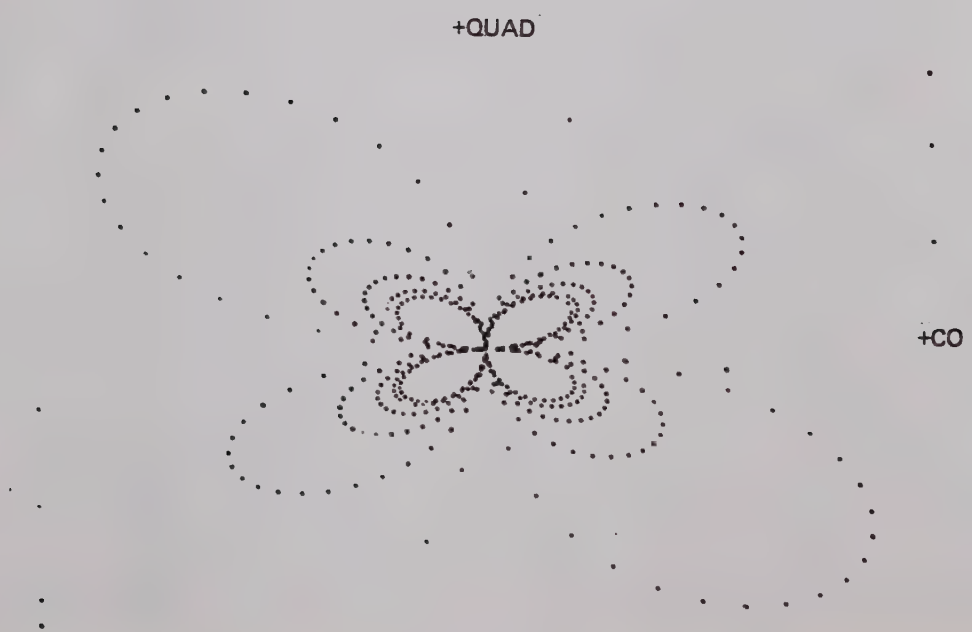


Figure 20. Function 2 – Nyquist Plot (Co vs Quad)

FUNCTION 3 – FORWARD TRANSFORM A(2048)

Function 3 generates 1024 frequency domain coefficients in either rectangular or polar coordinates from 2048 time domain data samples. Input data from Channel A signal input is alternately loaded into input memory A and input memory B. Input data is multiplied by the selected weighting function and transferred to the processor memories. Data located in input memory A is transferred to processor memory real and data loaded in input memory B is transferred to processor memory imaginary. A forward transform is performed, coordinate conversion as selected by the display controls is made and the result is stored in output memory. Figure 21 gives the data flow for Function 3.

$S_x(j\omega)$ = Forward Transform

$S_x(j\omega) = a_n + jb_n$

$$a_n + jb_n = \frac{1}{N}\sum_{\kappa=0}^{N-1} Y(\kappa)W^{n\kappa}$$

where $W = e^{-j\frac{2\pi}{N}}$

$n = 0, 1, 2, \ldots\ldots, \frac{N}{2}-1$

n = Spectrum Component Number

$\kappa = 0, 1, 2, \ldots\ldots, N-1$

κ = Sample Number

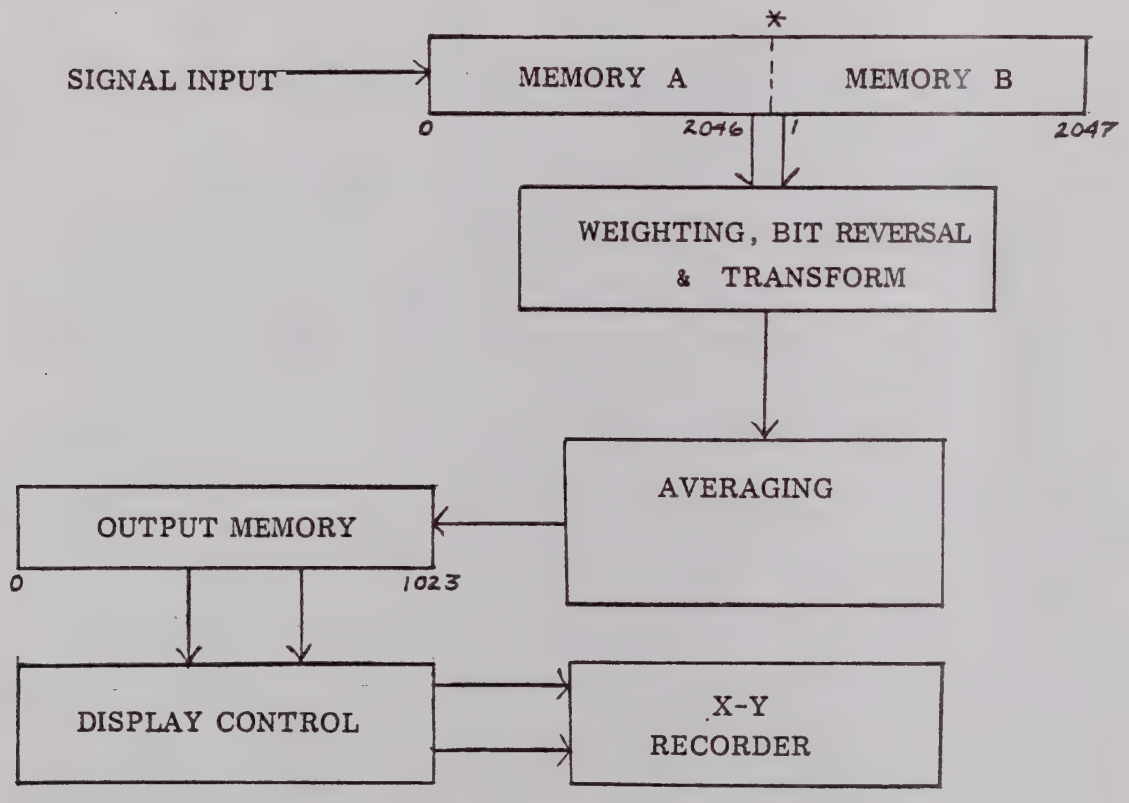


Figure 21. Data Flow For Function 3

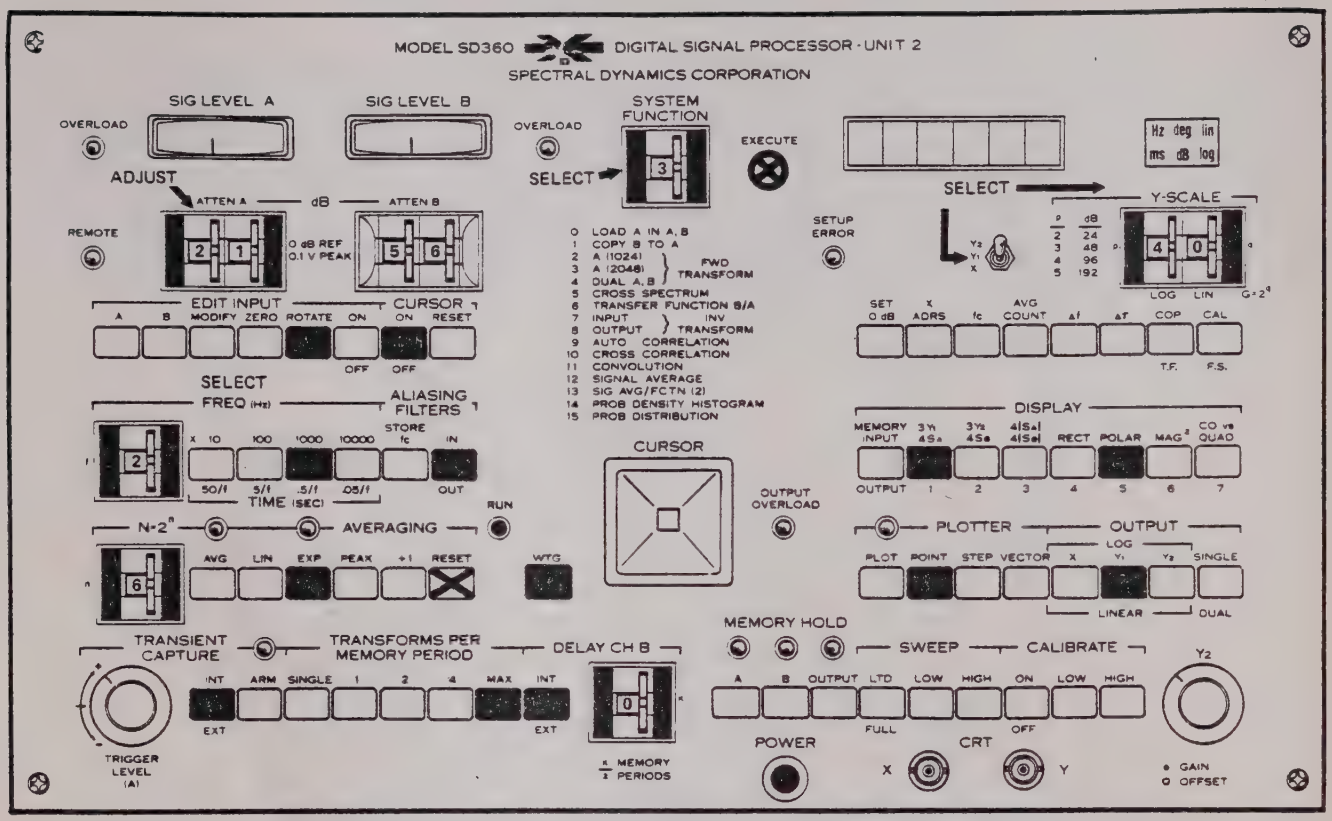


Figure 22. Function 3 Diagram

FUNCTION 3 – CH A 2048 POINT FOURIER TRANSFORM

1. Using the Function 3 Diagram, select the front-panel controls designated ■ and set the thumbwheel switches to the number X as designated by ⊠.

2. Apply the data signal to CH A input, select the desired Frequency Range and adjust the ATTEN A maximum SIG LEVEL A without OVERLOAD.

3. Momentarily push the control designated ⊠. Push EXECUTE ⊗.

The X-Y display will be a single trace of the unaveraged spectrum of the data signal displayed as LOG amplitude versus LINEAR frequency. (See Figure 23.) The procedure to identify peaks of interest on the spectrum is the same as given in Function 2 Diagram, Step 4.

FUNCTION 3 – CH A 2048 POINT FOURIER TRANSFORM WITH AVERAGING

The procedure for averaging the Function 3 data is the same as in the Diagram for Function 2 – with Averaging.

FUNCTION 3 – DISPLAY OPTIONS

1. To display phase data from Function 3 select DISPLAY POLAR and $3Y_2 4S_B$ buttons and release OUTPUT LOG Y1 button. The display represents the phase of the incoming data signal with respect to the internal sine/cosine table and the phase angle corresponding to the cursor location will be displayed on the LED readout with the axis toggle switch in the Y1 position. An attempt to average phase information in this mode will lock up the display.

2. To display Linear Amplitude Squared data in Function 3, select DISPLAY $3Y_1 4S_A$ and MAG^2. The OUTPUT LOG Y1 button should be released. Changes in the output gain are accomplished by adjusting the q switch. Each increment of the q switch will either increase or decrease the output gain by a factor of 2.

3. Display of the real component of Function 3 is accomplished by selecting DISPLAY $3Y_1 4S_A$ and RECT push buttons. Display of the imaginary component is accomplished by selecting $3Y_2 4S_B$ and RECT push buttons (See Figures 24 and 25). The OUTPUT LOG Y1 button should be released for these rectangular displays.

4. Selecting the axis toggle switch Y2 position for any of the Function 3 displays will cause the SETUP ERROR light to come on.

Figure 23. Function 3 – Polar Log

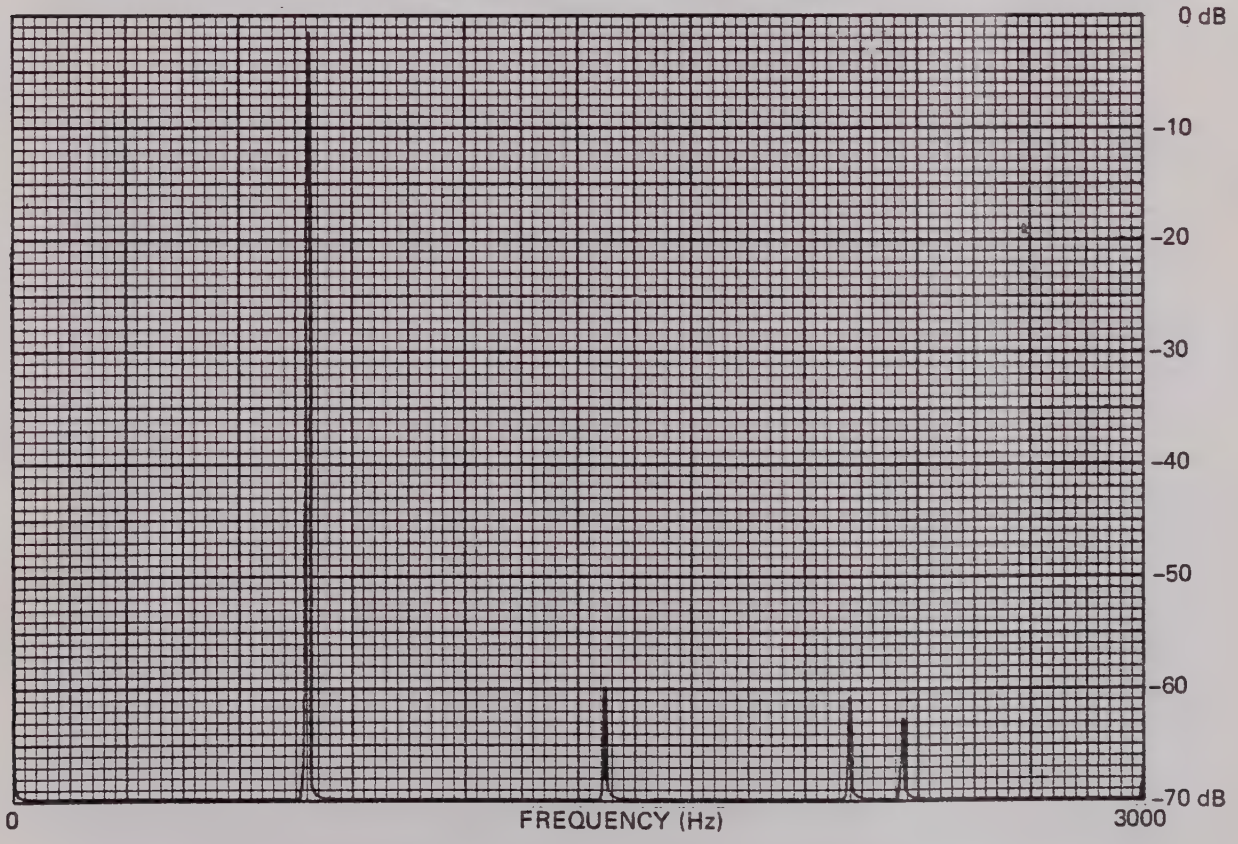


Figure 24. Function 3 — Rectangular

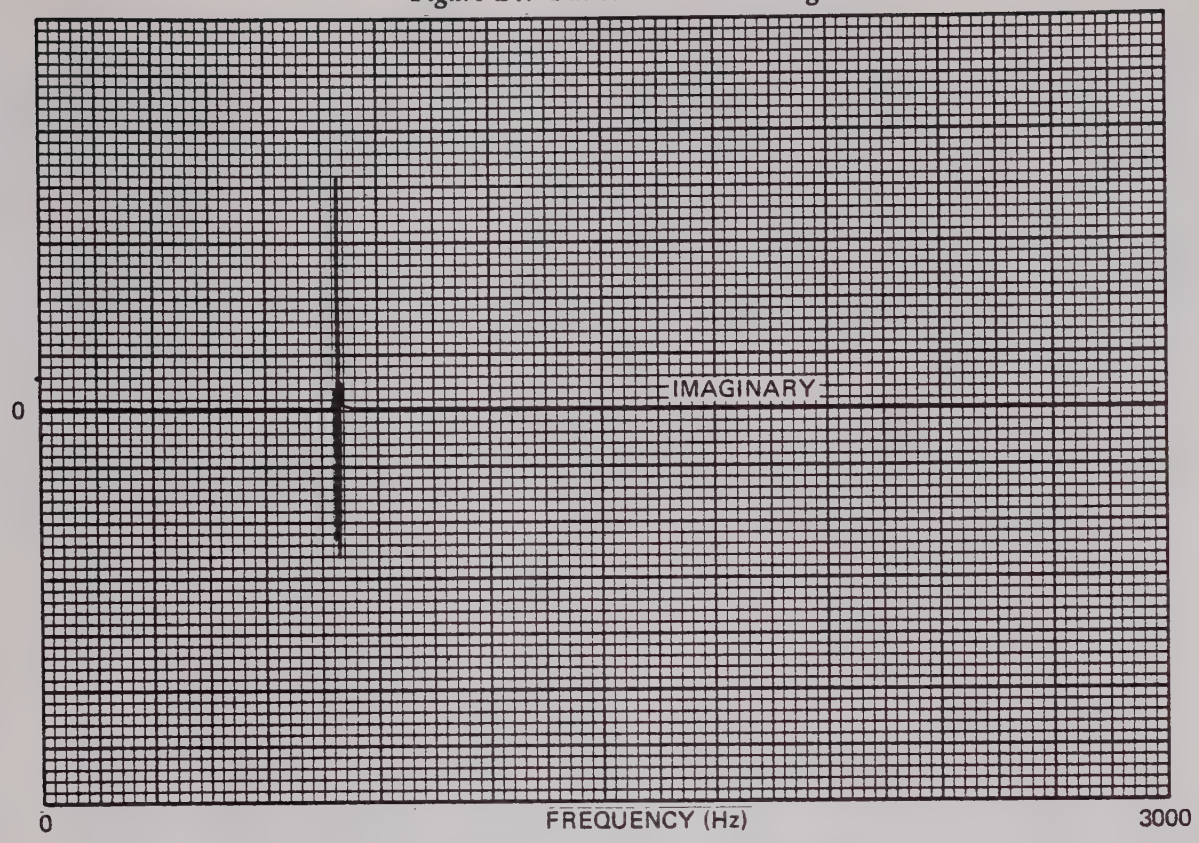
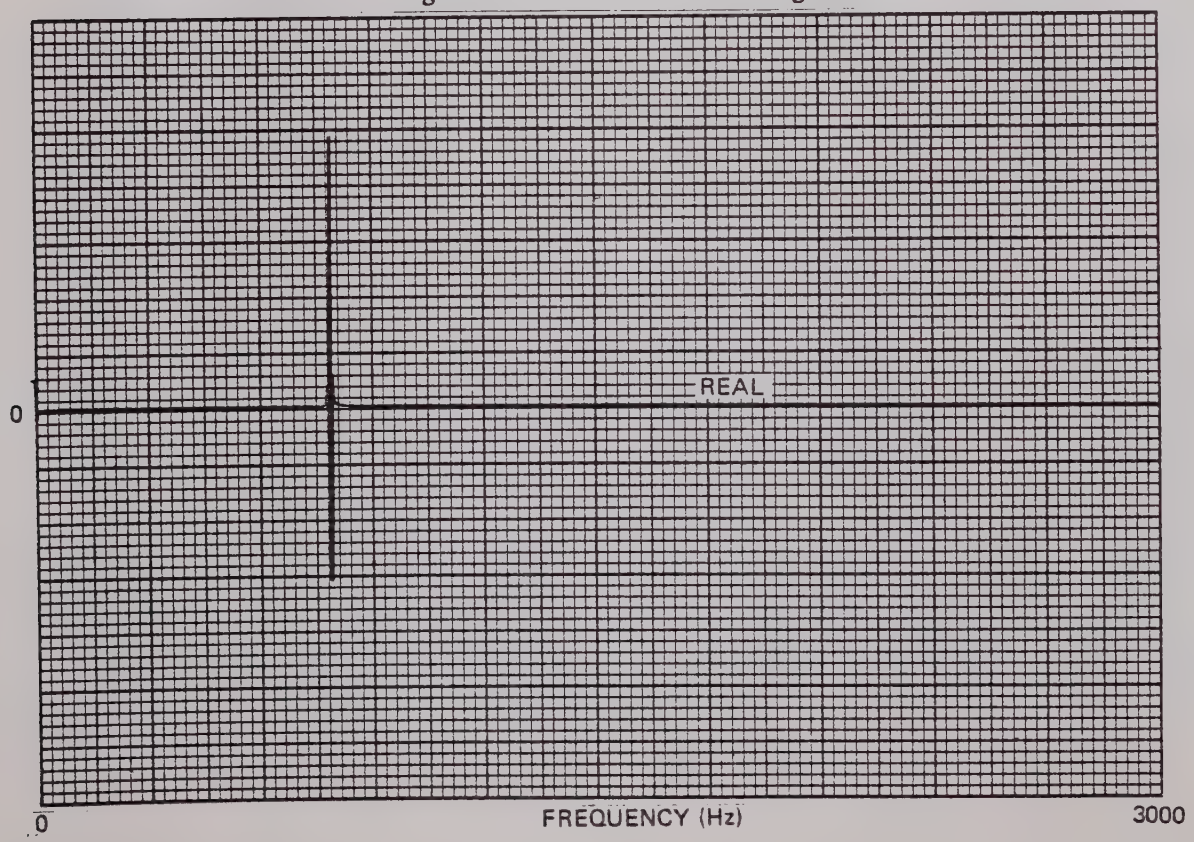


Figure 25. Function 3 — Rectangular



FUNCTION 4 – DUAL FORWARD TRANSFORM A, B

System function 4 generates 512 frequency domain coefficients in either rectangular or polar coordinates for each input data channel.* The 1024 input data samples from Channel A are placed in memory A and the 1024 input data samples from Channel B are placed in memory B. When an execute pulse occurs, the data samples are taken from memory A, multiplied by the selected weighting function and placed in processor memory real. Simultaneously, the data samples are taken from memory B, multiplied by the selected weighting function and placed in processor memory imaginary. A dual forward transform is performed on the data, coordinate conversion as selected by the display controls is made and the result placed in output memory. Figure 26 is the data flow for Function 4.

$$\left.\begin{aligned} S_x(j\omega) &= a + jb \\ S_y(j\omega) &= c + jd \end{aligned}\right\} = \text{Same definition as functions 2 and 3.}$$

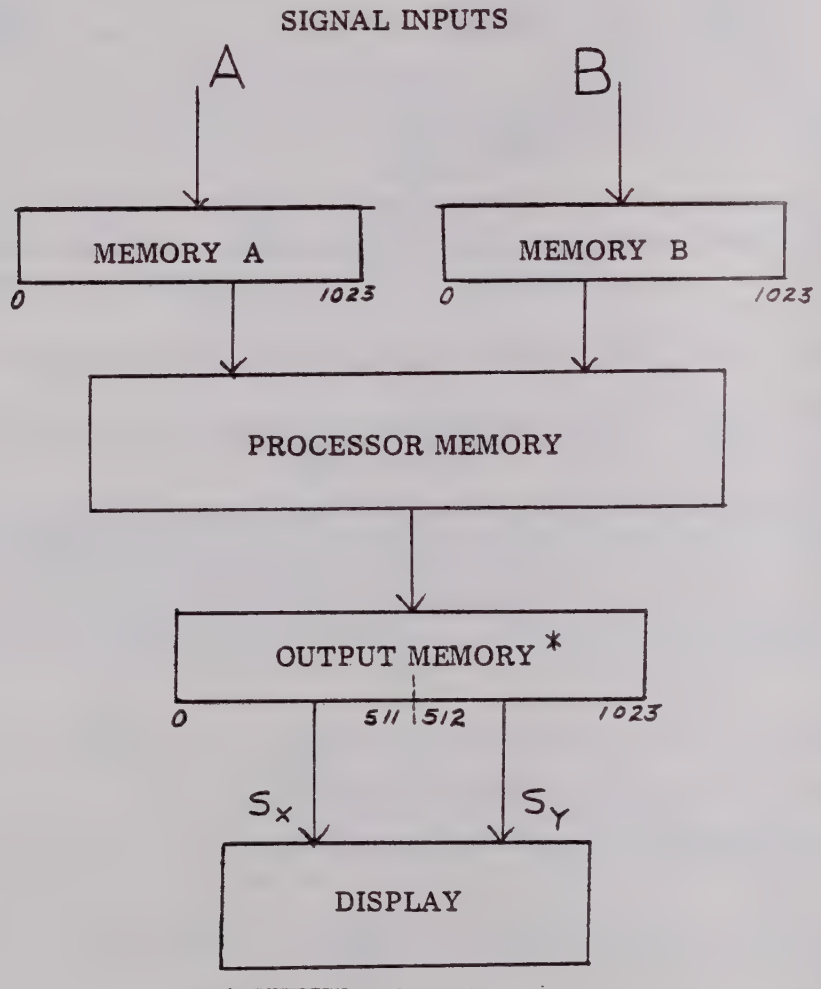


Figure 26. Data Flow For Function 4

*Output memory may be loaded with either $S_x(j\omega)$, $S_y(j\omega)$, or $|S_x(j\omega)|$, $|S_y(j\omega)|$.

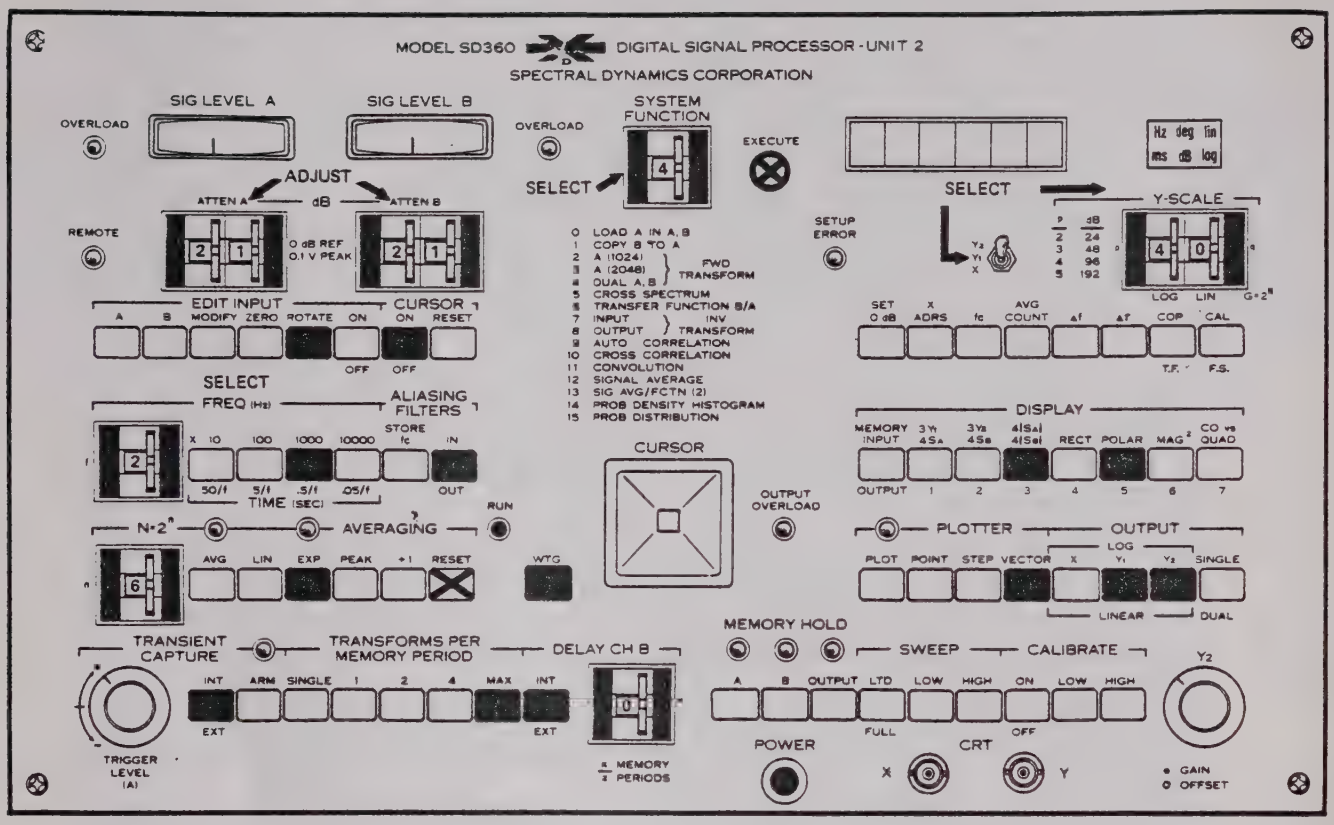


Figure 27. Function 4 Diagram

FUNCTION 4 – CH A AND CH B DUAL 1024 POINT FOURIER TRANSFORM

1. Using the Function 4 Diagram, select the front-panel controls designated ■ and set the thumbwheel switches to the number designated by ■.

2. Apply data signals to CH A and CH B inputs and select the desired Frequency Range. Adjust the ATTEN A and B controls for maximum SIG LEVEL A and B without OVERLOAD.

3. Momentarily push the controls designated ⊠. Push EXECUTE ⊗. The X-Y Display will be a dual trace of the unaveraged spectra of the input signals displayed as LOG amplitude versus LINEAR frequency. *

   a. Place the Cursor dot on a peak of interest on the lower trace (Y1 axis).

   b. Perform steps a. through f. of Function 2 Diagram for Y1.

   c. Change the axis toggle switch to Y2. Place the Cursor dot on a peak of interest on the upper trace (Y2 axis). The Cursor dot tracks the axis selected by the axis toggle switch.

   d. Repeat steps a. through f. of Function 2 Diagram, substituting Y2 for Y1.

*If the X-Y Display does not appear to be updating, or appears frozen in function 4, check that both LOG Y1 and Y2 push buttons are both in, or out, if DISPLAY 4 $|S_A|$ 4 $|S_B|$ push button is selected.

4. Change the format for the display of CH A output data to view magnitude and phase by selecting Display $3Y_1 4S_A$ and releasing the OUTPUT LOG Y2 push button. The lower trace (Y1) will be magnitude with LOG amplitude. The upper trace (Y1) on the display will now be phase with LINEAR amplitude.

5. Change the Format for the display of CH A output data to view the real and imaginary parts of the analysis by selecting the DISPLAY RECT and releasing the OUTPUT LOG Y1 push button.

6. Change the format of the output data for CH B to view the magnitude and phase by selecting Display $3Y_2 4S_B$ and pushing the OUTPUT LOG Y1 push button. The upper trace again will be phase with LINEAR amplitude and the lower trace will be magnitude with LOG amplitude.

7. Change the format for the display of output data for CH B to view the real and the imaginary parts of the analysis by selecting the DISPLAY RECT and releasing the OUTPUT LOG Y2 push button.

FUNCTION 4 – DUAL 1024 POINT FOURIER TRANSFORM WITH AVERAGING

The procedure for Averaging in Function 4 is the same as in Function 2.

FUNCTION 2, 3, 4, AND 13 – POWER SPECTRAL DENSITY ANALYSIS (PSD)

Power spectral density analysis (PSD) is a very useful tool to determine characteristics of stationary random signals. The PSD plot shows the distribution of energy for random data in the signal as a function of frequency. The total area under the PSD curve is the mean squared value of the signal in the plotted frequency range of the signal.

When PSD is performed digitally, advantage is taken of the relationship between the Fourier transform of a function and its power spectrum at that frequency. If the Fourier spectrum of a signal is defined as $S_x(j\omega) = a_n + jb_n$ then the power spectrum, $G_{xx}(\omega)$, at that frequency is:

$$G_{xx}(\omega) = S_x(j\omega)\ S_x^*(j\omega) = (a_n + jb_n)(a_n - jb_n) = a_n^2 + b_n^2 .$$

In practical implementation of this formula $G_{xx}(\omega) = \frac{1}{2} S_x(j\omega)\ S_x^*(j\omega)$.

Because the signal is random in nature, an estimate of the true PSD is given by averaging the results of several ensembles. The PSD value must be averaged and not the Fourier transforms themselves since the real and imaginary parts along with phase will tend to average to zero.

The basic method of implementing the PSD formula $G_{xx}(\omega) = S_x(j\omega)\ S_x^*(j\omega)$, digitally follows:

1. Take N samples of data
2. Do a Fourier transform of the data signal
3. Multiply the Fourier transform of the signal by its complex conjugate (*).
4. Add this power spectrum ($a^2 + b^2$) to the previous sum of power spectra stored in a memory location
5. Repeat desired number of times (N)
6. Divide ΣPS by the total number (N) and display results by selecting MAG^2 display control.

Distribution of this estimate about the true value follows a Chi-square distribution with the statistical confidence of the data estimate increasing as the number of ensembles increases. However, the improvement in the PSD estimate is a converging function of the number of ensembles included; generally, a value of N between 64 and 512 is considered adequate.

THIS PAGE INTENTIONALLY
LEFT BLANK

THIS PAGE INTENTIONALLY
LEFT BLANK

Figure 28. Rectangular Weighting

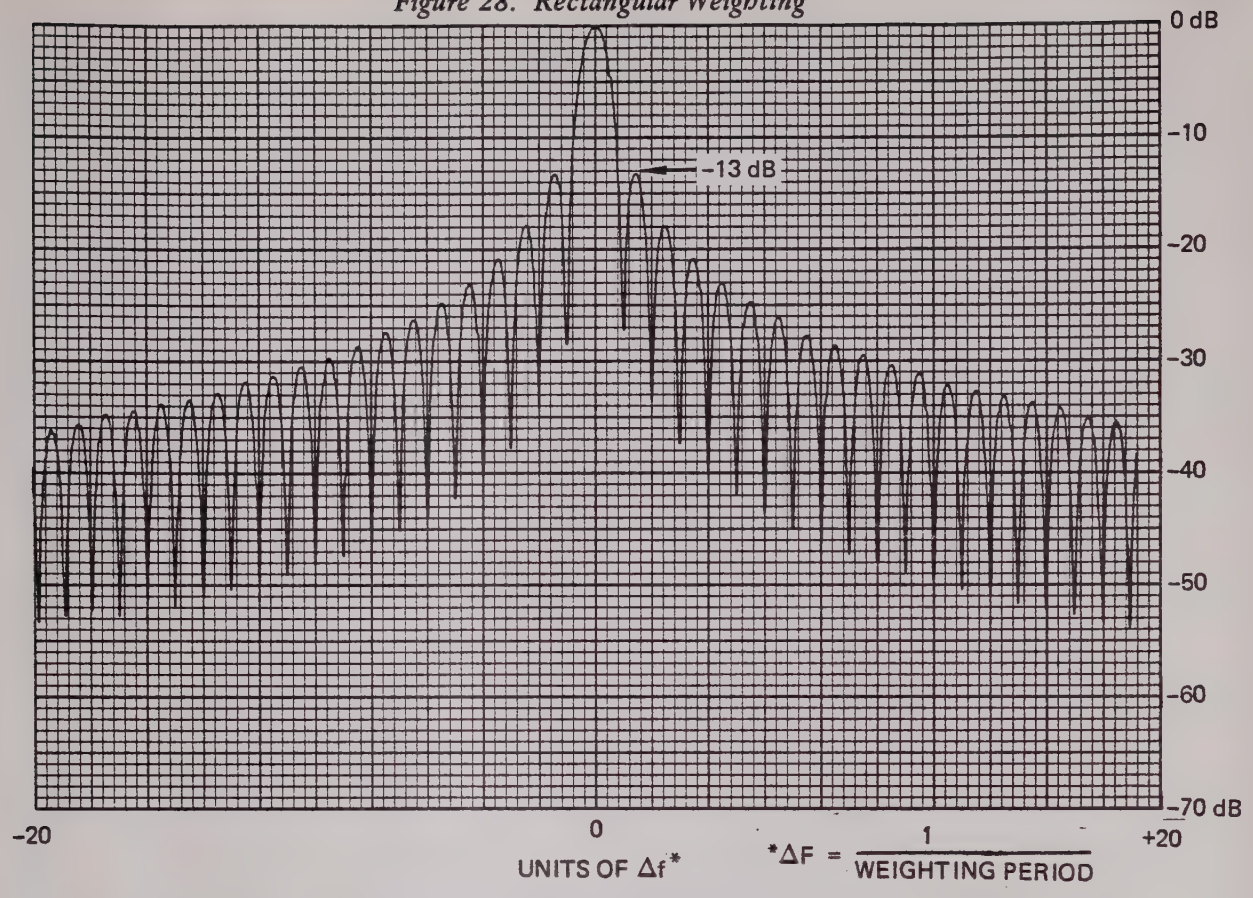


Figure 29. Hanning Weighting

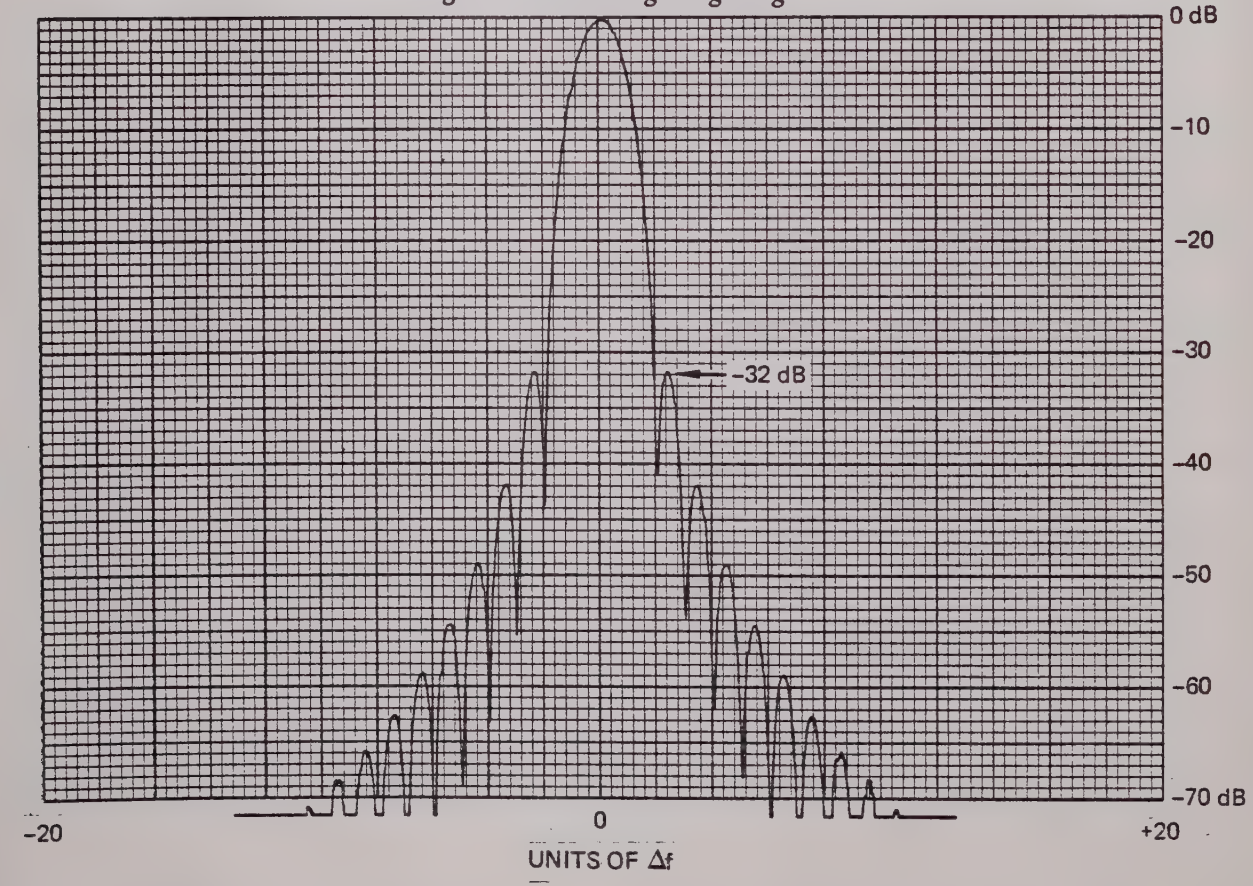


Figure 30. Gaussian Weighting

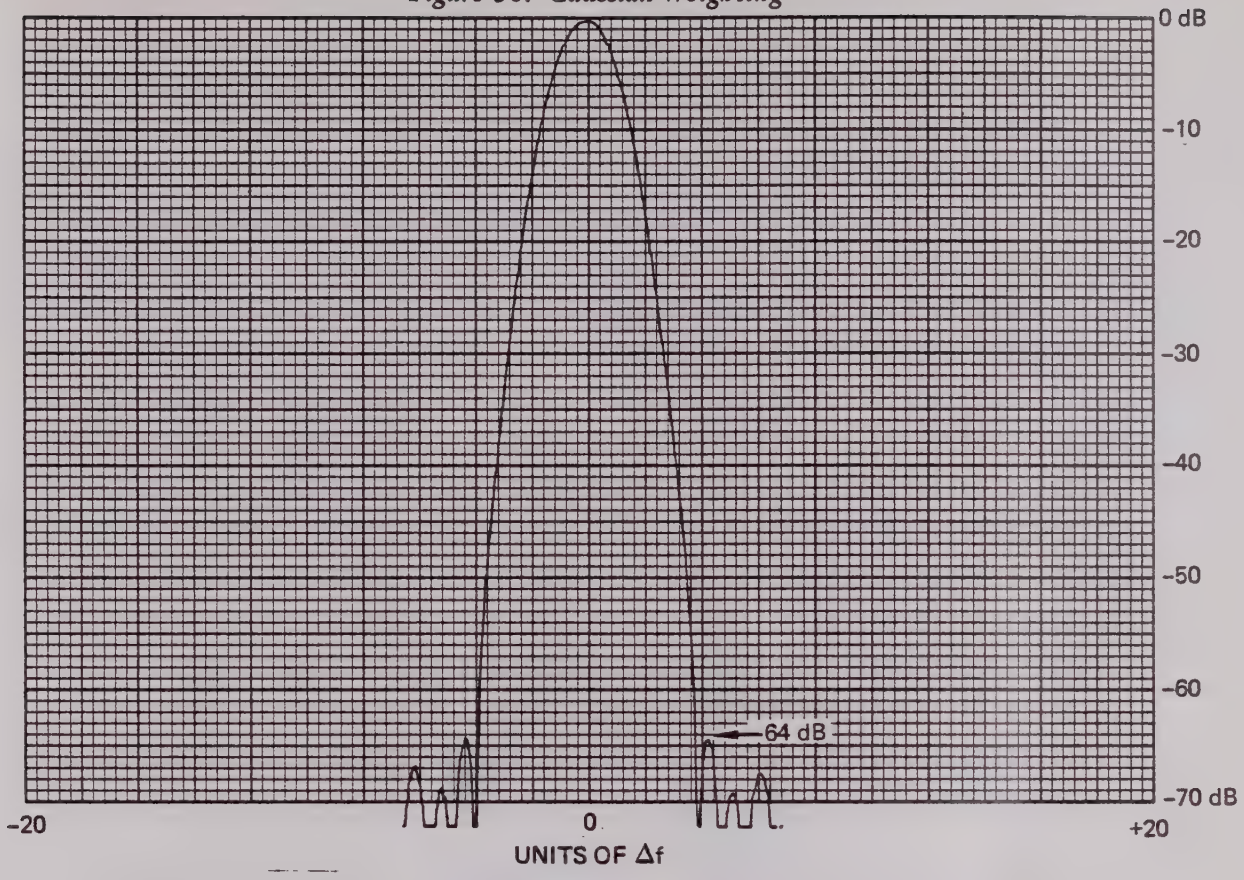
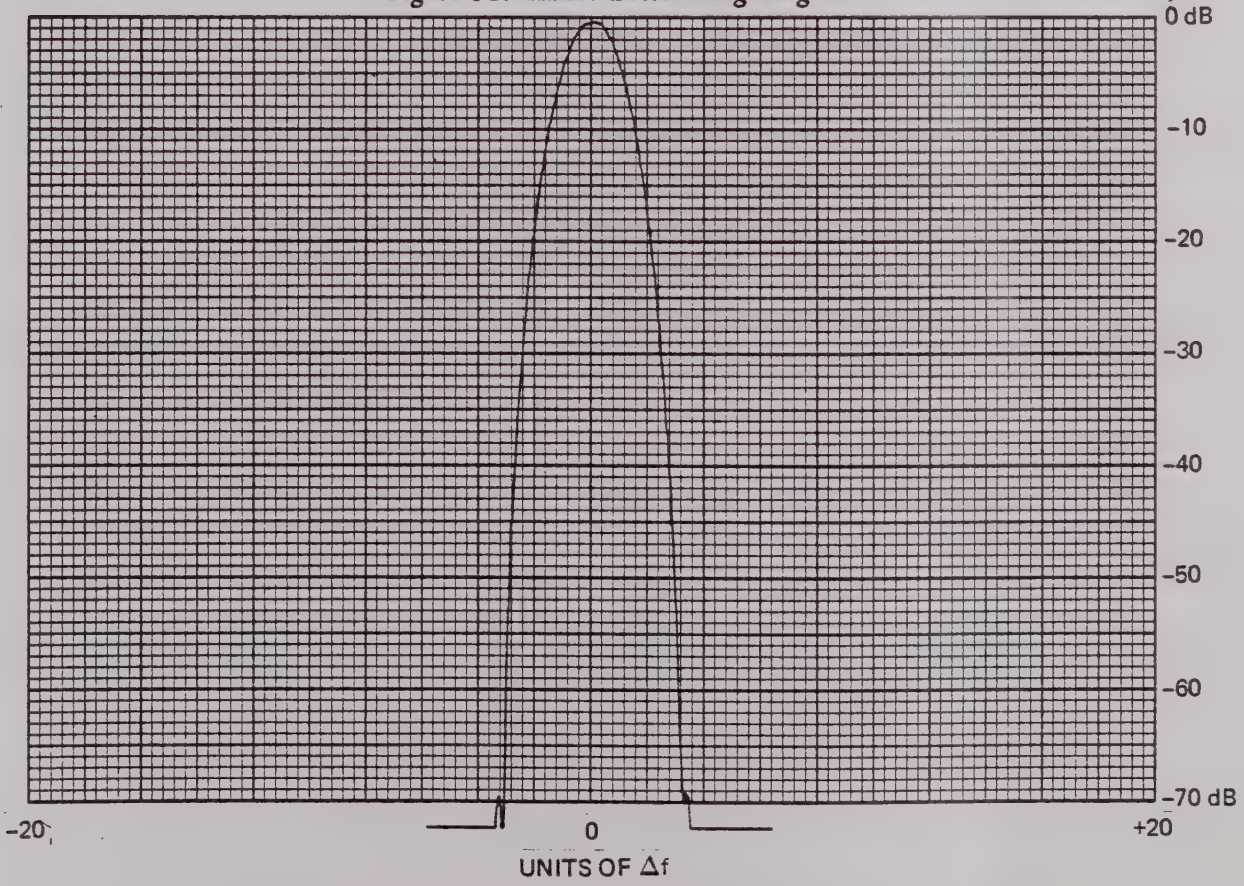


Figure 31. Kaiser-Bessel Weighting



CALIBRATION FOR POWER SPECTRAL DENSITY ANALYSIS

The SD360 can obtain the power spectrum of a signal using Functions 2, 3, 4, or 13, and displays the results as Log Amplitude vs Frequency, or Magnitude Squared vs Frequency; Magnitude Squared denoting the square of the Linear Amplitude. To obtain an estimate of the power spectral density (PSD) function requires that the display device used, either an oscilloscope or X-Y recorder, be calibrated in units of volts^2/Hz. The normalization of the power spectral density function to a 1 Hz unit is necessary to eliminate the effect that the analysis bandwidth on each analysis range contributes to the overall energy that passes through the filter. For the SD360, the PSD calibration relationship is:

$$\frac{\text{Volts}^2}{\text{Hz}} = \frac{(V_{rms})^2}{\text{Noise BW}} \qquad \text{Equation (1)}$$

The voltage referred to in Equation 1 can be normalized to reflect the sensitivity of the signal transducer used; e.g., 10 mV/g or 100 mV/ips. Thus, any selected full scale input attenuation can be scaled directly in engineering units instead of volts. The effective noise bandwidth on the SD360 is determined as follows:

For Gaussian Weighting $BW_{eff} = 1.84\,\Delta f$

For Kaiser-Bessel Weighting $BW_{eff} = 1.78\,\Delta f$ Equation (2)

Where Δf is the filter spacing for the corresponding analysis range.

Substituting Equation 2 into Equation 1 yields the following for Gaussian Weighting.

$$\frac{\text{Volts}^2}{H_z} = \frac{(V_{rms})^2}{1.84\,\Delta f} \qquad \text{Equation (3)}$$

As an example, consider that the PSD of a single data signal is desired using a 2 kHz analysis range and 20 dB attenuation on Channel A. This input attenuation setting corresponds to a 1 V peak full scale input signal. Since a calibration tone is used, the rms value is 0.707 times the full scale peak voltage selected. If the data signal is processed using a 1024 point transform, the Δf spacing on the 2 kHz range is 4 Hz. Substituting these values in Equation 3 yields the following relationship.

$$\frac{\text{Volts}^2}{H_z} = \frac{(1.0 \times 0.707)^2}{1.84\,(4.0)} = 0.068$$

This represents the PSD calibration value, simulated by a full scale tone on the 2 kHz analysis range using 20 dB of attenuation. Calibration values for other analysis ranges can be determined using Equation 3.

The amplitude of the fundamental of the internal calibration signal can be used to represent the full scale PSD calibration values. The cursor can be positioned on top of the peak of the fundamental and the span of the display device can be set up using the input attenuators on the SD360. The ZERO control on the X-Y recorder is used to establish the 0 dB or full scale PSD level on the X-Y graph paper.

An external tonal calibration signal can be used to establish an absolute PSD calibration level. The rms value of the input tone can be measured and the corresponding noise bandwidth used for the analysis range that is selected to determine the PSD calibration level using Equation 3.

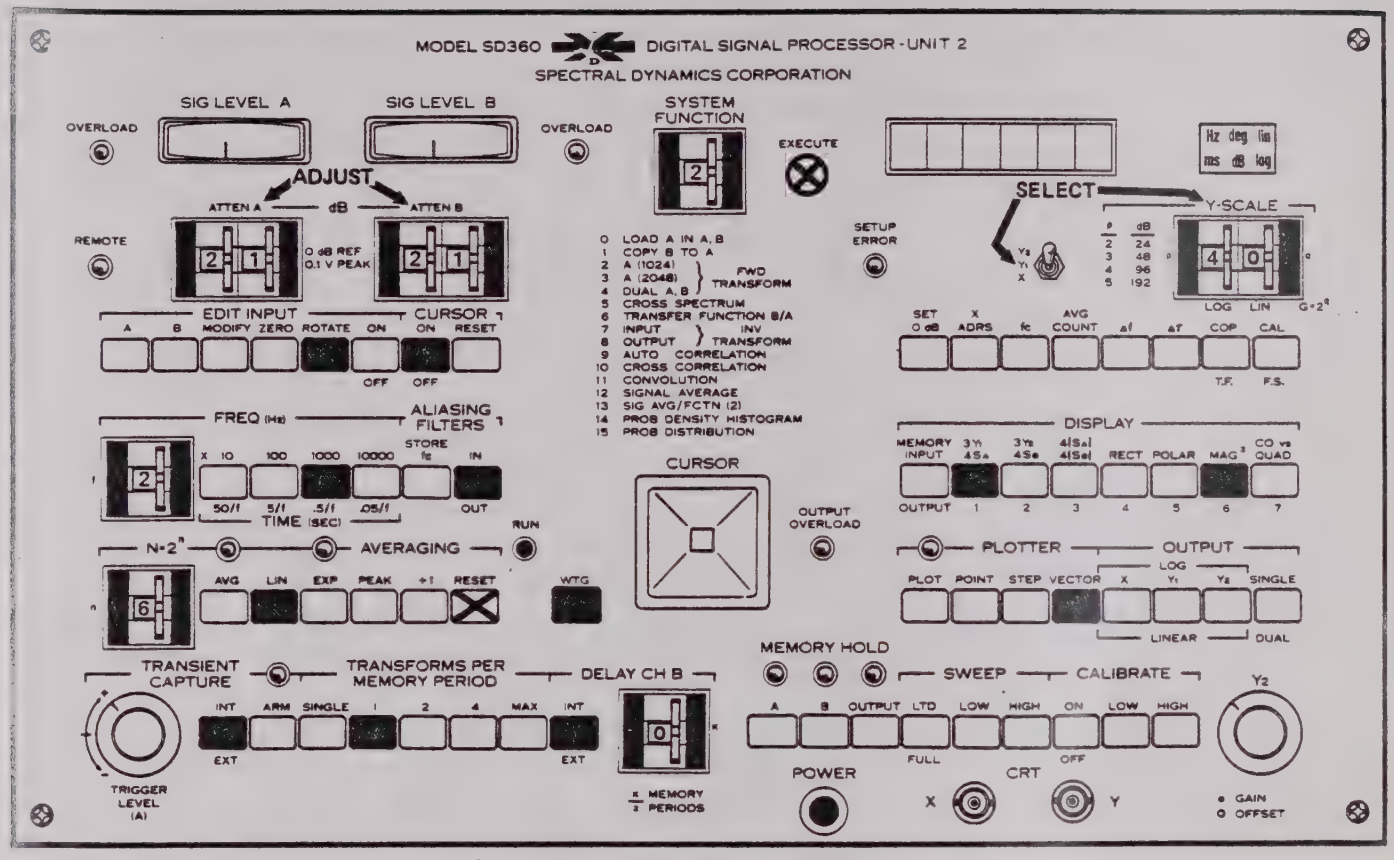


Figure 32. Power Spectral Density Analysis Diagram

THIS PAGE INTENTIONALLY
LEFT BLANK

FUNCTION 5 – CROSS SPECTRUM

The cross-spectrum between two signals indicates similarities between two signals in the frequency domain and includes phase information. The magnitude of the cross-spectrum at each frequency is the product of the magnitudes of the Fourier transforms of the two signals with a phase equal to the phase difference of the two transforms. The resultant phase is dependent upon the order of the cross-spectrum, G_{yx} or G_{xy}.

$$|G_{yx}(j\omega)| = |S_y(j\omega)|\,|S_x(j\omega)|$$

$$\theta_{yx}(j\omega) = \theta_y(j\omega) - \theta_x(j\omega)$$

$$G_{yx}(j\omega) = G^*_{xy}(j\omega)$$

The cross-spectrum is computed the same as a dual forward transform; however, the results are transferred to the output memory as the real and imaginary rectangular coordinates. These rectangular coordinates are used to compute magnitude and phase for display, if desired, and may also be averaged since effectively a phase difference is being taken between the two signals which does not require a fixed reference in time. Figure 35 gives the data flow for Function 5.

$$G_{yx}(j\omega) = S_y(j\omega) \cdot S^*_x(j\omega)$$

$$G_{yx}(j\omega) = \text{Co Spectrum} + j\,\text{Quad Spectrum}$$

$$|G_{yx}(j\omega)| = \left(\overline{Co_{yx}(\omega)}^2 + \overline{Quad_{yx}(\omega)}^2\right)^{1/2}$$

$$\theta_{yx}(j\omega) = \tan^{-1}\left(\frac{Quad_{yx}(j\omega)}{Co_{yx}(j\omega)}\right)$$

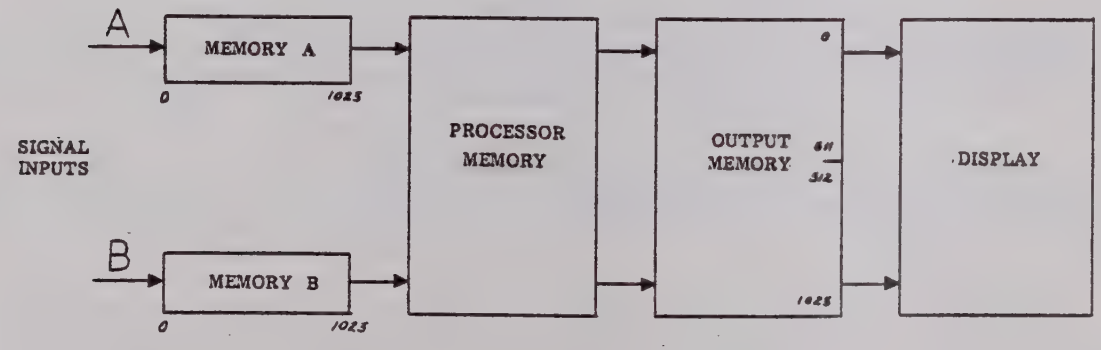


Figure 35. Data Flow For Function 5

*denotes complex conjugate

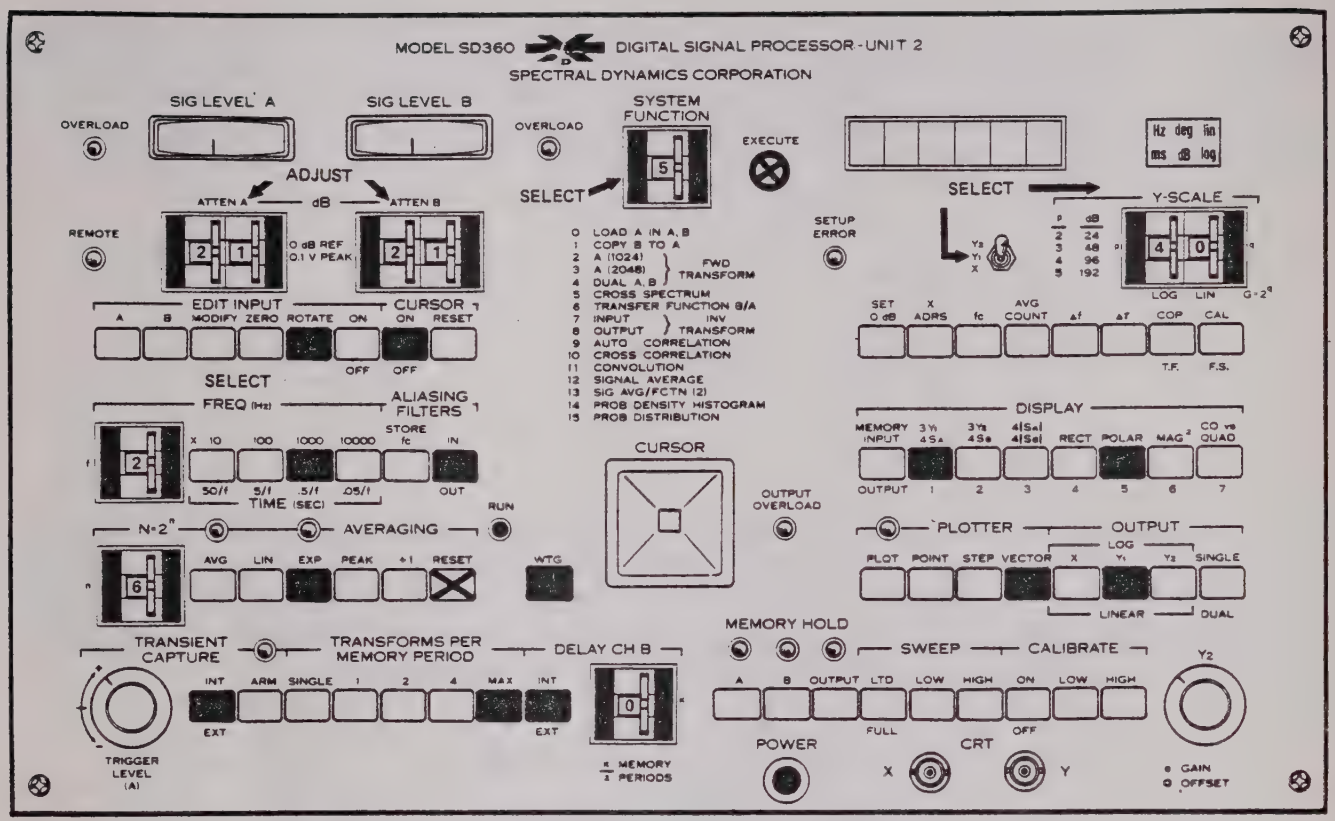


Figure 36. Function 5 Diagram

FUNCTION 5 – CROSS SPECTRUM

1. Using the Function 5 diagram, select the front panel controls designated ■ and set the thumbwheel switches to the number X designated by [X].

2. Apply data signals to Channel A and Channel B inputs and select the desired frequency range. Adjust the attenuator A and B controls for maximum signal level A and B without overload.

3. Momentarily push the control designated ⊠. Push EXECUTE, ⊗. The X-Y display will be the unaveraged cross spectra of the input signals displayed as log amplitude versus linear frequency in the lower trace and the upper trace, the corresponding phase angle between the two input data signals.* (See Figures 37 and 38).

---

* When viewing a cross-spectrum in terms of LOG Magnitude, a choice is available for scaling for Y_1 display. Cross-spectrum can be defined in terms of cross-power which gives a log scaling of cross-power (dB) = 10 log P_2/P_1.

Cross-spectrum also represents the product of 2 transforms (one a complex conjugate) and can be presented in terms of $VOLTS_2$, giving a scale factor of "20 log" or 20 dB per decade.

For a cross-power display with a "log 10" scaling select POLAR LOG Y_1.

For a voltage squared display with "20 log" scaling, select MAG^2 LOG Y_1. The MAG^2 display produces a scaling identical to that for F2, F3, F4, with POLAR LOG Y_1 selected.

4. A selected peak of interest in the data may be precisely defined by using the Cursor control and observing the respective amplitude and frequency in the six-digit LED readout.

5. The cross-spectrum can also be viewed in rectangular coordinates as the real and imaginary components. This is done by selecting DISPLAY RECT and releasing the LOG Y1 push button. (Figure 39).

FUNCTION 5 – CROSS SPECTRUM WITH AVERAGING

In Function 5, the real and imaginary components can be averaged since effectively a phase difference is being taken between the two signals which does not require a fixed reference in time.

1. Averaging in the Cross Spectrum function is performed by selecting the desired number of averages on the N thumbwheel switch and pushing the AVG push button. The Averaging light will be on when the selected number of averages has been accumulated in the LIN mode. If Exponential or Peak modes were selected the Averaging light will flash on and off.

2. After a completed average is obtained, a flicker-free X-Y display can be observed by selecting the SINGLE TRANSFORMS PER MEMORY period push button. This stops the Processor and the RUN light on the front panel will go out. The data contained in the output memory will be continuously displayed in a flicker-free fashion.

3. To change the display format of the data stored in the output memory, select the desired format and push the EXECUTE button. The output data will automatically be re-formatted and presented for X-Y display. (Figure 40).

CROSS SPECTRUM DELAY

Delay Channel B Feature – This control is used for transient capture of two independent signals, cross-spectrum processing, or cross-correlation work. In two-channel processing, applications exist where the similarities or mutual properties between two signals do not coincide with one another until some time delay has elapsed. Conceivably, this time delay could exceed the corresponding memory period for the analysis range that is used for processing the input data signals. Channel B data can be delayed in one-half memory period steps up to 7½ memory periods until the coincident portion of Channel B data is loaded and available for cross-spectrum processing with Channel A data.

If the 7½ memory period delay is not long enough, an external TTL pulse can be applied to connector J7 of SD360-Unit 1 to start loading data upon external command. The INT push button on the front panel of the SD360-Unit 2 must be released for Unit 1 to recognize this external command.*

* The EXT delay control is designed to operate in a transient capture sense providing one load of MEMORY A and one delayed load of MEMORY B.

FUNCTION 6 – TRANSFER FUNCTION B/A

The transfer function, $H(j\omega)$, which provides information about the transfer of a signal through a system, is the cross-spectrum of the input and output signals normalized by the input power spectrum. The transfer function presents the same data as a cross-spectrum in a somewhat different form. The magnitude of a transfer function is the ratio of the magnitudes of the input and output signal Fourier transforms. The phase angle of the transfer function equal to the difference in phase of the two signals.

$$H(j\omega) = \frac{G_{yx}(j\omega)}{G_{xx}(\omega)} = \frac{S_y(j\omega)}{S_x(j\omega)} = \frac{S_y(j\omega)}{S_x(j\omega)} \cdot \frac{S_x^*(j\omega)}{S_x^*(j\omega)}$$

With function 6 selected and transfer function selected on the front panel control, COP/T.F., a dual Fourier transform is performed on the input signal data in memory A and on the output signal data in memory B. A power spectrum $G_{xx}(\omega)$ for the input signal is computed and placed in an additional output memory. After coordinate conversion is performed as selected by the display controls, the real and imaginary components of $G_{yx}(j\omega)$ are placed in output memory where they are usually averaged before display. If a Function 6 is performed, the transfer function or the coherent output power can be displayed by selecting the proper display controls. Figure 41 is a typical Inertance Transfer Function Display.

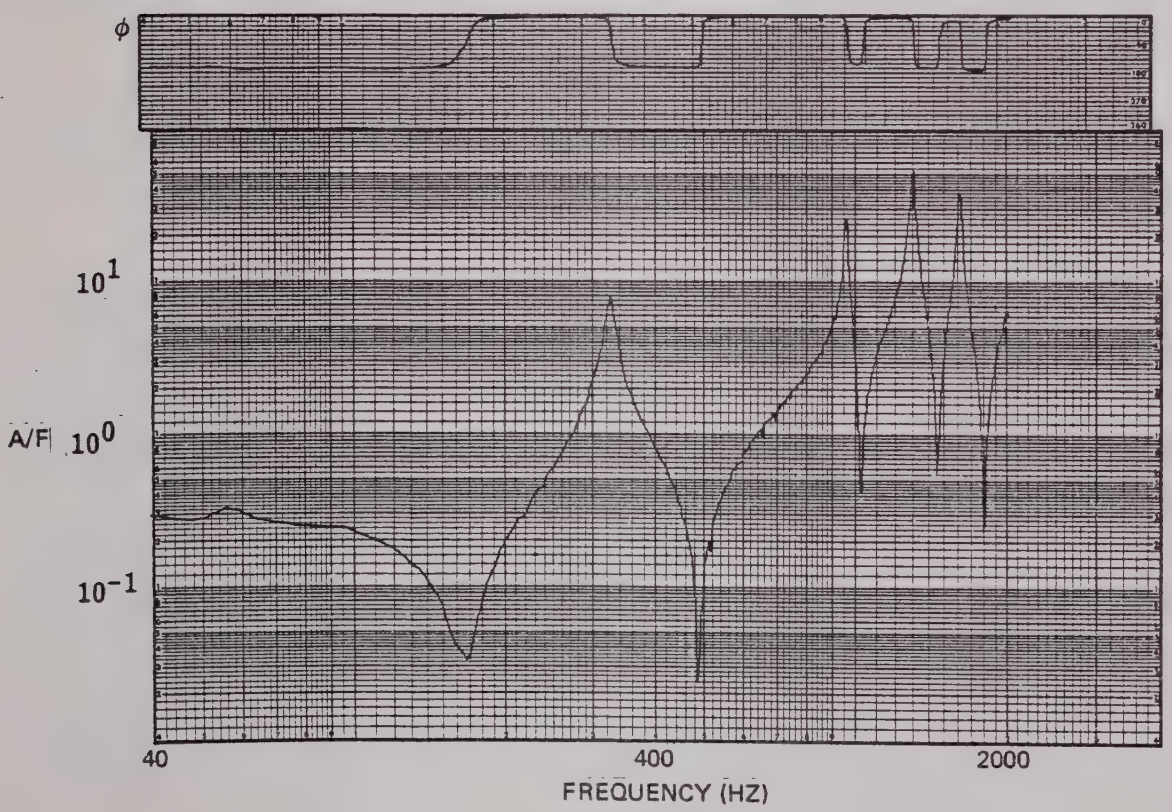


Figure 41. Transfer Function B/A

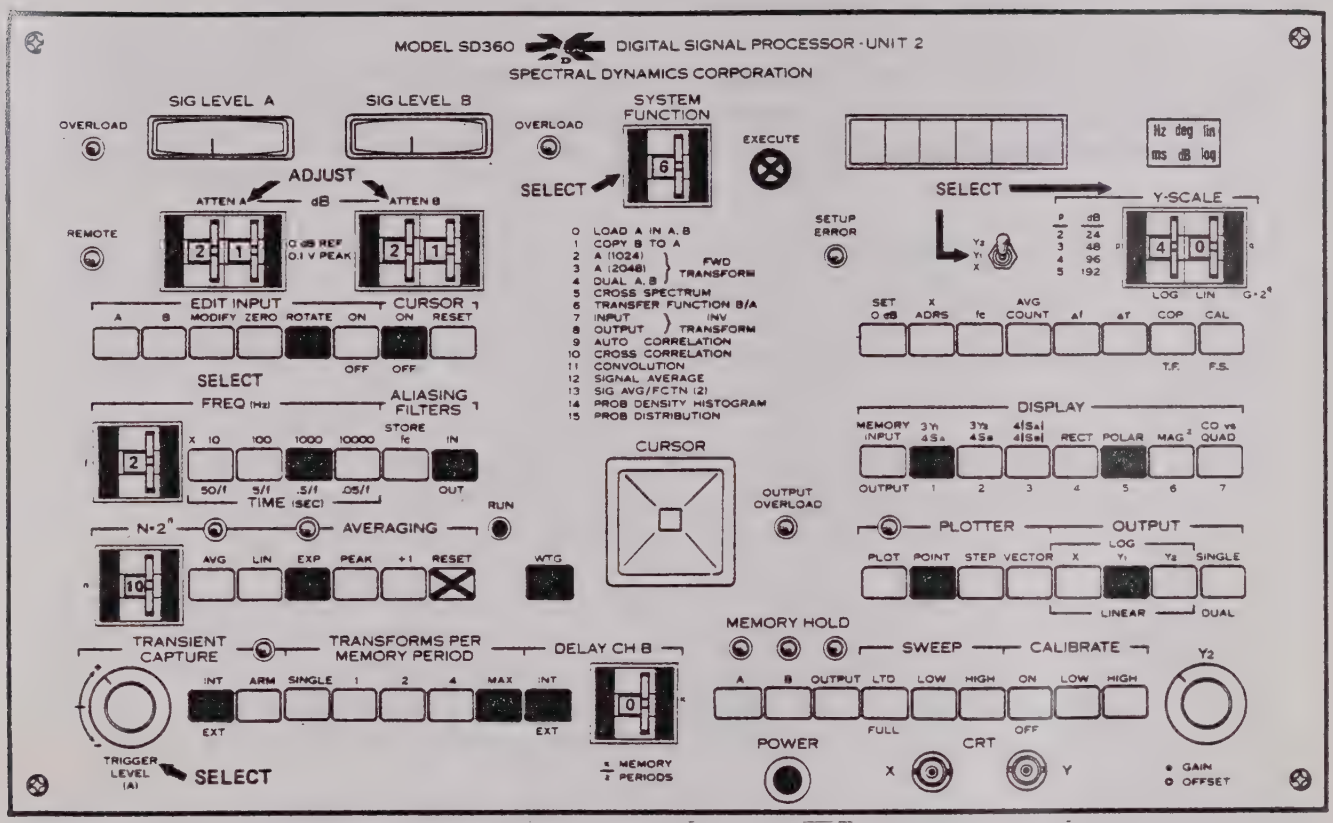


Figure 42. Function 6 Diagram

FUNCTION 6 – COHERENT OUTPUT POWER (COP)

The coherent output power (COP) represents the portion of the output power produced by a given source in the presence of incoherent noise. Coherent output power is computed by multiplying the output power spectrum, $G_{yy}(\omega)$, by the coherence function, γ^2 of the output spectrum and a selected input signal. The coherence function is defined as:

$$\gamma^2 = \frac{\overline{|G_{yx}(j\omega)|}^2}{G_{xx}(\omega)G_{yy}(\omega)} \leq 1$$

$$TF = \frac{G_{yx}(\omega)}{G_{xx}(\omega)}$$

Therefore,

$$COP = \frac{\overline{|G_{yx}(j\omega)|}^2}{G_{xx}(\omega)G_{yy}(\omega)} \cdot G_{yy}(\omega) = \frac{\overline{|G_{yx}(j\omega)|}^2}{G_{xx}(\omega)}$$

$$COP = \frac{\overline{|G_{yx}(j\omega)|}^2}{G_{xx}(\omega)} \cdot \frac{G_{xx}(\omega)}{G_{xx}(\omega)} = \frac{\overline{|G_{yx}(j\omega)|}^2}{\overline{|G_{xx}(\omega)|}^2} \cdot G_{xx}(\omega)$$

$$COP = \overline{|H(j\omega)|}^2 \cdot G_{xx}(\omega) .$$

$$\text{ALSO } COP = \gamma^2 \cdot G_{xx}(\omega)$$

If the input and the output signals are completely uncorrelated, the coherence function will equal 0. If they are correlated at all frequencies, the coherence function will equal 1. If the coherence function is greater than zero, but less than unity, one or more of three possible situations exist.

1) Extraneous noise is present in the measurements.
2) The system relating input and output is not linear.
3) The output is an output due to other inputs as well as the chosen input source.

When the front panel control C.O.P. is selected in combination with function 6, the coherent output power is computed from the values stored in output memory as a result of the Fourier transforms performed on the input and output signals as discussed in function 6, transfer function.

Function 6 - Transfer Function B/A
(For use with SD360-18 which contains the Coherence Function Option.)

The Model SD360-18 Digital Signal Processor is a standard instrument to which has been added the capability of measuring and displaying the normalized coherence function. The coherence function is typically used in conjunction with a measured transfer function in order to qualify the measurement.

Ideally, a transfer function measurement is performed on two signals; one representing a unique system forcing function, the other representing the measured system's response. In the absence of any extraneous noise, if no other responses are present and if the system being analyzed is linear, the transfer function measurement should be excellent. However, since the basic cross-spectrum measurement does not give a causal relationship, it has been found useful to define a measurement qualifier. This parameter, called the coherence function, tries to identify how much of the measured cross-spectrum is related to the measured input and output power spectra. The coherence function $\gamma^2_{yx}(\omega)$ is defined in Appendix 6-A.

Earlier it was seen that many different types of waveforms can be used to excite a system over its full frequency range. Figure 1 is a typical driving point inertance transfer function calculated using the FFT approach and a random forcing function. In this case, 128 averages were taken, although a good transfer function measurement can be achieved with fewer averages; in some cases, as few as 16 averages.

The coherence function corresponding to this transfer function measurement is seen in Figure 2. Several frequencies of apparently low coherence are evident from this display. During this type of analysis, there are several reasons why the coherence function may appear to be quite low. These are enumerated in Appendix 6-A.

In the Model SD360-18, additional microprogramming has been added to permit the computation of the normalized coherence function as described above. Also, storage memory has been added to accommodate loading the averaged output power spectrum G_{yy}, which is necessary as a basic part of the coherence function computation. With this option, it is possible to take a selected number of ensemble averages of input/output relationship and then choose the function to be displayed using the push button configuration chart labeled, "Operating Instructions for Coherence Function." For example, once a number of spectra have been averaged using Function 6, it is then possible to choose one of the following displays.

a. The input and output power spectra [$G_{xx}(\omega)$ and $G_{yy}(\omega)$];

b. The cross-spectrum [$G_{yx}(\omega)$] in terms of polar or rectangular coordinates;

c. The transfer function [$H(j\omega)$] in terms of rectangular or polar coordinates including phase;

d. The transfer function and the coherence function [$\gamma^2_{xy}(\omega)$]; or

e. The coherent output power $[\gamma^2_{xy}(\omega)G_{yy}(\omega)]$ and phase.

A typical operating sequence follows.

1. Select Function 6 and press the EXECUTE control;

2. Select MAX TRANSFORMS PER MEMORY PERIOD;

3. Press the AVERAGING RESET control;

4. Select the desired frequency range of operation;

5. Adjust the input ATTEN A and B controls according to a known sensitivity position, from an externally applied calibration signal or such that no input overload is detected;

6. Select LIN AVERAGING mode;

7. Select a suitable number of averages, such as 64 (set n=6);

8. Press the AVG control;

9. When the LED above AVG is out and the LED above EXP is on, in a steady fashion, press the SINGLE TRANSFORMS PER MEMORY PERIOD control;

10. Select the desired display configuration. For example, to display transfer function and coherence function select DISPlay controls 2 and 5. Note that for correct coherence function displays, the q switch must be set to 7.

When using Function 6, such that a selected number of averages is to be performed and then data is to be recorded by an X-Y recorder, it is always a good idea to select SINGLE TRANSFORMS PER MEMORY PERIOD after the selected number of averages has been reached. When switching from one mode of display to another in conjunction with the operating instructions for coherence function, note that the EXECUTE control must be pressed once after each new display has been selected.

Calibration of the X-Y Recorder for Function 6

Since many different functions can be computed and displayed under the basic Function 6 operation, usually it will be necessary to set, establish or check the calibration conditions each time a new display is selected. A typical display combination in Function 6 is transfer function and phase using polar coordinates with log transfer function magnitude and linear phase. For this combination, display controls 1 and 5 are selected.

Assume that linear-linear draft paper (e.g., K&E type 46-0707) is to be used. Scale selection will then be linear abscissa (10 divisions), log ordinate (4 divisions, 20 dB/division) and linear phase (2 divisions, 180°/division). The calibration sequence will be X axis, Y1 axis, and then Y2 axis. Note that this calibration procedure can be performed even after data has been averaged and stored in the processor memory.

In the diagrams that follow, the calibration controls and corresponding recorder position are shown. Assume a frequency range of 5 kHz, a p switch setting of 4, and a q switch setting of 7. For X-axis calibration, the calibration control in the lower righthand corner of Unit 2 front panel is pressed. For linear X-axis operation, the CALIBRATE LOW push button is always left IN. With just the LOW push button pressed, as shown in Diagram 1, a frequency reading of 0 Hz is shown and the plotter ZERO control is adjusted so that the pen rests in the lower lefthand corner of the recording paper. By pressing both CALIBRATE LOW and HIGH push button controls simultaneously, no vertical pen deflection is provided, but the pen now moves horizontally to the full scale frequency of 5000 Hz, which is also indicated in the LED readout. This is seen in Diagram 2. For linear X-axis calibration, the CALIBRATE ON and LOW controls are always left IN. Alternately pressing and releasing the CALIBRATE HIGH control, causes the X axis to receive a full horizontal swing command. Alternating back and forth between button positions of Diagrams 1 and 2 is then done until the proper ZERO and GAIN controls for the recorder have been established.

For Y1 calibration, the CALIBRATE HIGH and ON controls are left in. Alternately pressing and releasing the CALIBRATE LOW control then furnishes a voltage change of exactly 80 dB (for p = 4) which can be used to establish the zero and gain settings of the recorder vertical axis. For the example given, 80 dB is made to represent four major divisions (40 minor divisions) of the graph paper. If a different sensitivity is required for the vertical Y1 scale, such as 10 dB/major division, a different calibration technique is necessary since it may not be possible to place the entire 80 dB dynamic display range on the expanded graph paper. Typically, under these conditions, the calibration signal must be passed through the input attenuators A and B by setting CAL/F.S. control in the upper righthand corner of the SD360 Unit 2 front panel to the CAL position (IN). Normally, this control is left OUT (Full Scale position). Note that if the signal is to be passed through the input attenuators, a real time mode of operation is needed. Any data in the Processor output memory must be erased. The cursor is then placed over the fundamental component of the calibration signal and the attenuator typically reduced in 10 dB steps, adjusting the recorder gains and offsets accordingly.

For Y2 calibration, use the control settings shown in Diagrams 5 and 6. The axis-toggle switch is placed up to the Y2 position and now the Y2 GAIN and OFFSET controls in the lower righthand corner of SD360 Unit 2 front panel are used. At this point, the GAIN and OFFSET controls of the recorder should not be changed; only the Y2 GAIN and OFFSET controls of the DSP should be used. Pressing the CALIBRATE ON and CALIBRATE HIGH controls gives a phase indication of +179.9 degrees. This

is then used to set the recorder maximum Y2 deflection. Pressing the CALIBRATE LOW control, as seen in Diagram 6, then establishes a 0 degree condition used to set the recorder to the middle of the phase swing. Note that the SD360 phase output is always set to display between -180^{o} and $+180^{o}$, but the calibration signals cover half this range, from 0^{o} to $+180^{o}$. Alternately pressing and releasing the CALIBRATE LOW push button control while leaving the CALIBRATE HIGH control in, then gives a 180^{o} swing from 0^{o} to $+179.9^{o}$ and the Y2 GAIN and OFFSET controls are used accordingly to establish this swing on the X-Y recorder.

A second common display group would be that of log frequency-log transfer function magnitude and linear coherence function. Th calibration controls for this combination are given in Diagrams 7 through 12. For LOG X calibration, the CALIBRATE LOW settings give an indication of 50 Hz while the CALIBRATE HIGH setting gives a reading in voltage proportional to 5000 Hz. Alternately switching back and forth between CALIBRATE LOW and CALIBRATE HIGH, the recorder X GAIN and OFFSET controls are then alternately used to establish the desired swing. Note that under these conditions, CALIBRATE LOW and HIGH should not be pressed simulatneously as this will give a Cursor readout and a voltage proportional to 4850 Hz. Note that an intermediate step in the X-axis calibration, such as 500 Hz, can be established using the limited sweep control. For example, to set the high sweep limit at 500 Hz, release the CALIBRATE ON push button and move the cursor horizontally until the indicator shows 500 Hz. Then press the SWEEP HIGH control.

During calibration modes of operation, when the CALIBRATE HIGH push button control is selected, giving an indication and voltage proportional to 5000 Hz, the output indication and voltage will be proportional to 500 Hz if the limited sweep control is then pressed. For the example given here in Diagrams 7 through 12, it is assumed that graph paper of two decades horizontally and 4 decades vertically, with an extra vertical section for the coherence function, is employed. Note that the LED readout during LOG Y1 calibration procedures is meaningless.

To calibrate the vertical axis for the coherence function, follow the sequence of Diagrams 11 and 12. Pressing either the CALIBRATE LOW control, or both the CALIBRATE HIGH and LOW controls, gives voltage output proportional to zero coherence value. Physically, this represents the same voltage output as -180^{o} for a phase presentation. By pressing only the CALIBRATE HIGH control, the LED readout and voltage output proportional to a coherence value of 1.997 is presented. Although this is a value which will never be reached in practice, it represents a voltage equivalent to that of $+180^{o}$ for a phase presentation. Thus if 2 vertical divisions of recording paper are used for phase presentations, the equivalent voltage swing for coherence displays will be one-half that of the phase presentation. Leaving the CALIBRATE HIGH control in the IN position and alternately pressing and releasing the CALIBRATE LOW control, will produce a voltage swing which is exactly twice that which will be achieved during practice. If appropriate recording paper is used, this permits rapid changeover from coherence to phase displays without necessitating a corresponding calibration change.

To display the input and output power spectra as part of a Function 6 display, the same calibration techniques are used as in Function 4. Similarly, to display a cross-spectrum presentation as part of a Function 6 operation, the same calibration techniques are employed as used in Function 5.

Measuring Coherent Output Power

Since coherent output power represents a measurement of the portion of a total power spectrum which is coherent with some reference, its measurement is usually conducted in conjunction with a measurement of total signal power. This means that the measurement of the coherent output power will normally be conducted either prior to or following the performance of a power spectral density function, which will then give us the power spectrum of both the total composite signal we are trying to interpret and a selected reference signal.

1. To perform a coherent output power measurement set the front panel controls according to the F6 (COP) diagram.

2. Select Function 6; press the EXECUTE button and be sure that the TF/COP control is in the COP position (IN).

3. Press the AVG RESET control.

4. Set the desired number of averages and LIN AVG (Expo averaging can alternately be used).

5. Select MAX TRANSFORMS PER MEMORY PERIOD.

6. Select MAG^2, LOG Y_1, and LIN Y_2.

7. Press the AVG control to begin accumulating the desired number of averages.

8. Once the desired number of averages has been accumulated such that the AVG light goes out, press the SINGLE TRANSFORMS PER MEMORY PERIOD control.

9. Set the p switch to zero; then return it to p = 4 prior to recording.

By setting the p switch to zero, all vertical roll has been removed from the coherent output power display when using a log magnitude. By selecting a MAG^2 display (Display Control 6), the cursor readings of selected components will retain the same calibration as seen in a normal power spectrum analysis, such as Function 4. Select Display Control 3 and 5/6 two-channel power spectrum analysis (Function 4) in conjunction with a coherent output power measurement. A typical total power spectrum analysis, together with a reference spectrum, would be such as seen in Figures 3 and 4,

Figure 5 shows the coherent output power spectrum and the coherence function for this set of conditions.* Note that in this case, the coherence function is extremely low at all frequencies except those at which the reference signal components are common to both channels.

When performing a coherent output power analysis, recall the equation

$$\text{coherent output power} = \gamma^2_{yx}(\omega)\, G_{yy}(\omega) = \frac{|G_{yx}(j\omega)|^2}{G_{xx}(\omega)}$$

is being used.

The measured coherent output power depends upon the level of the output power spectrum and the coherence between this power spectrum and the reference input, no upon the level of the reference input. For this reason, when measuring coherent output power where the total response signal is applied through Input B of the SD360 and the reference signal to Input A, the level of the reference signal on Input A is not critically important. Note that although coherent output power measurements can be made with any SD360, the actual measurement and display of the normalized coherence function requires the "-18" modification to the SD360.

* The plots in figure 5 were made by performing a Transfer function and Coherence sequence. The Coherence function was plotted. Then a Coherent Output Power and Phase sequence was performed, and the COP plotted.

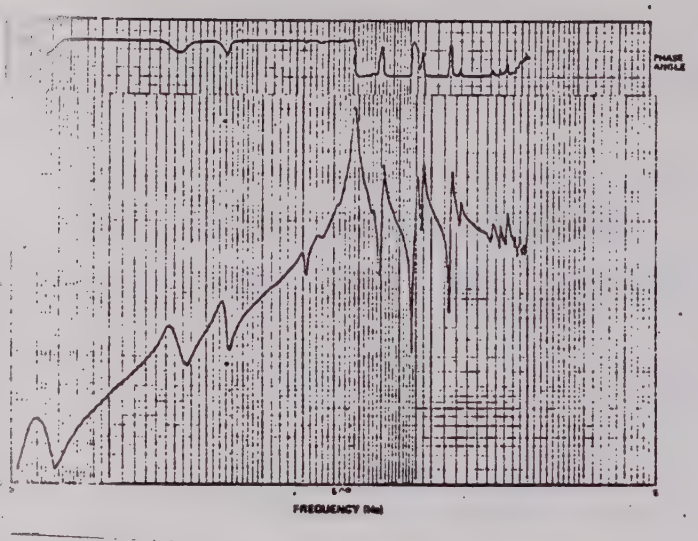


Figure 1. DRIVING POINT INERTANCE TRANSFER FUNCTION (128 AVERAGES)

SPECTRAL DYNAMICS CORPORATION OF SAN DIEGO

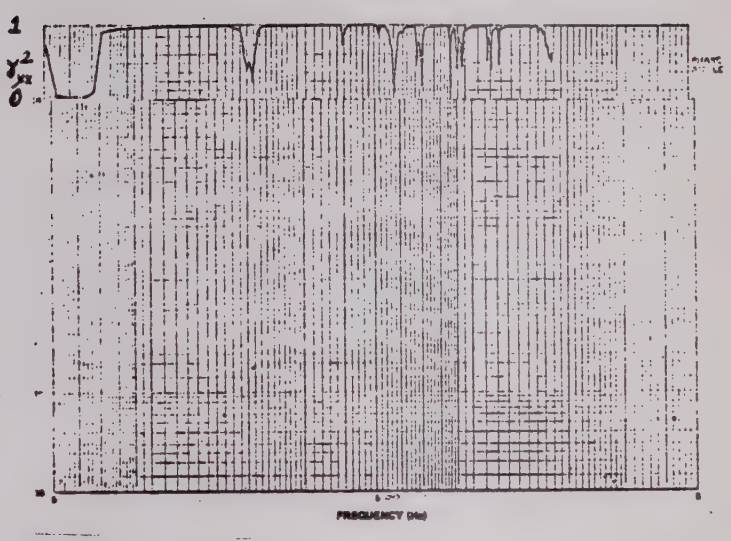


Figure 2. COHERENCE FUNCTION FOR FIGURE 1

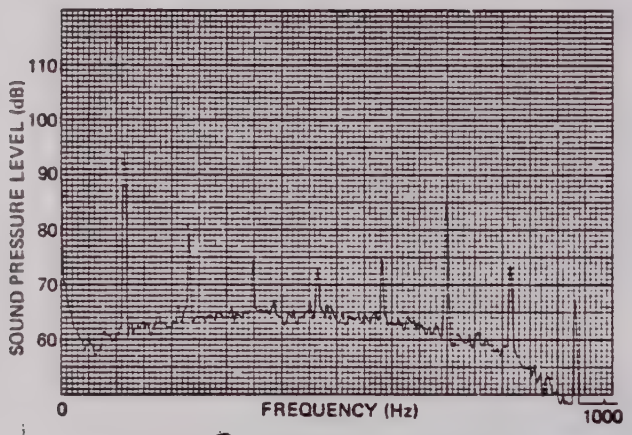


Figure 3. *Machine Noise — Location A*
4 Hz Bandwidth Analysis

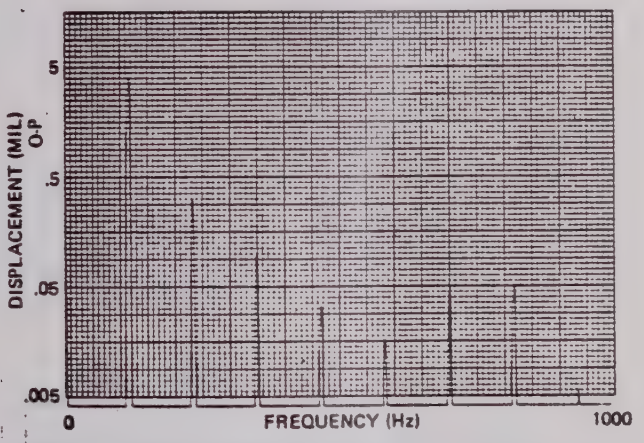


Figure 4. *Shaft Displacement Spectrum — Log Scale (7200 R.P.M.)*

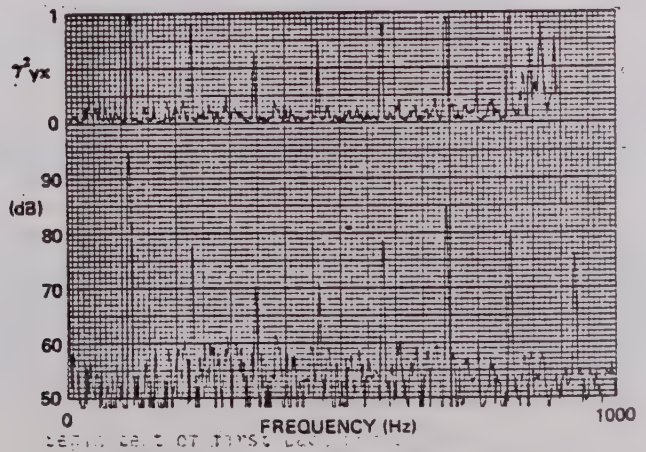


Figure 5. *Coherence Function Between SPLA and Shaft Motion*

APPENDIX 6A

The Measurement of Transfer Functions

Every structure and structural component responds to an input forcing stimulus in its own way. A structure may respond in one way to a static or slowly applied load, but exhibit much more severe response when placed under a dynamic load. If the vibration response of the structure is being measured, it is often difficult to distinguish between forced vibration inputs and input forces which might have been magnified or attenuated by the behavior of the structure itself. This is where the true power of frequency response testing lies. With the knowledge of a structure's dynamic characteristics, measured responses can be simply broken down by careful examination of the desired structural transfer function. In this way, calculations can be made to compute actual force level inputs during dynamic operation of a structure.

The transfer function or frequency response characteristic of a structure is a unique, repeatable parameter. Many techniques are used for transfer function measurements. Some involve removing components from a system, others can be used while a component or system is still in place, and still others are applicable to operating systems. The frequency response of a system or component can be described in any one of a number of ways with a wide variety of transducers employed to measure this unique characteristic. (Table 6A-I) In many cases, structural response measurements are made using a force gauge and accelerometer outputs. Normalized transfer functions can also be measured using the outputs of a) displacement probes or LVDT's; b) velocity transducers; c) strain gauges; d) microphones; e) current shunts; or f) Laser signals.

TABLE 6A - I

Force/Acceleration	=	Dynamic Mass
Force/Velocity	=	Mechanical Impedance
Force/Displacement	=	Dynamic Stiffness
Displacement/Force	=	Compliance
Velocity/Force	=	Mobility
Acceleration/Force	=	Inertance

Frequency response measurements can be made at a point of excitation (driving point transfer function), or can represent the response of a second point due to a driving input force at some other point (cross transfer function). For driving point force and acceleration measurements, either separate transducers can be employed, or one transducer can be used which contains both a force gauge and an accelerometer in one integral unit (mechanical impedance head). Regardless of the type of transducer used, it is important that the force gauge be mounted as closely as possible to the actual point of excitation and that a good bond be assured between the force transducer and the actual structure under test.

Since either a cross or driving point transfer function is attempting to describe a frequency response at a specific point, or between two points on a structure, it is important that the force be applied as much as possible to a point on the structure, as opposed to being applied over an area of the structure.

To measure the "true" frequency response function implies several factors. Among these should be included the following:

a. The measurement system, including transducers, has been completely calibrated;

b. The system is being operated over a linear region of its behavior, or at least non-linearities have been taken into account;

c. The system is being excited at a known point and response motion is being measured also at a defined point;

d. Unwanted distortion components have been eliminated from the measurement;

e. Unwanted or extraneous mass in the transducer and attachment systems has been ellminated.

No matter which transfer function is chosen for display, complex vectors are involved in the calculation. That is, the transfer function based on a forcing input and a response motion, consists of both a magnitude and a phase, or a real and an imaginary part. Figure 64-1 shows that typical transfer functions vary over tremendous dynamic ranges. The electronic instrumentation for measuring and displaying dynamic impedance characteristics must be capable of making these measurements in a repeatable, accurate manner.

Figure 64-2 shows a general instrumentation concept for performing complete transfer function measurements on an automatic basis. A sweep oscillator and force servo are used together with a small electrodynamic or hydraulic vibration exciter to drive the structure under test. Driving point measurements are made using the integral impedance head shown beneath the structure. Transfer compliance or impedance recordings are made using a roving accelerometer which also can be used for gathering mode shape data. Signal conditioners, such as charge amplifiers, then process the signals to the transfer function analyzer. The analyzed results in terms of magnitude, phase and frequency can then be recorded on a two-pen recorder or digitized for automatic data logging.

Due to the dynamic characteristics of structures being tested, the analysis system must be capable of detecting drastic changes in response levels from a structure. Because the dynamics of the structures to be tested can be tremendous, a wide and continuous dynamic range in instrumentation is required. In order to display this large continuous dynamic range, it is necessary to present the transfer function data in terms of a

log amplitude ratio. Since a wide frequency range is to be covered as well, and since many structures respond almost in a constant Q fashion, data is often presented in terms of a log frequency display. Naturally, the phase data will be presented on a linear basis.

It is necessary to be able to detect the fundamental levels of the response signal without being influenced by unwanted noise and distortion that is almost always present in this type of measurement. The definition of true frequency response implies the measurement of a dynamic characteristic on a frequency by frequency basis. Thus, the response information measured should be the data representing the response only at the frequency of the structure's excitation. No harmonic distortion or noise component should be included in the measured results. Some type of filtering technique is necessary in order to achieve accuracy in the transfer function analysis.

One of the most important measurement requirements is to obtain good phase data. In many cases, particularly in determining how a structure actually deforms, phase information is just as important as magnitude data. In addition to being part of the total transfer function picture and describing structural deformation and mode shapes, phase data can also reveal whether an apparent peak in a transfer function display is a driving point resonance or represents reflected motion from some other point on the structure.

In order to completely describe the behavior of the test structure, one of the final pieces of information needed, in addition to several transfer functions, will be the normalized mode shapes of the structure at each of its major resonances. Before beginning a modal analysis, the classic definition of a mode shape should be reviewed. When a structure is dynamically excited, its modes of vibration are invariant. That means that, unless by chance the structure is excited right at a nodal point, the structure's resonant frequencies will be excited.

Ideally, the structure should be excited into pure modal behavior at each of its resonances. It is not always possible to excite a pure mode although it is possible to get a very close estimate of pure modal behavior even though the structure may not deform into a pure mode of response. The frequency response of a structure is defined as

$$\text{TRANSFER FUNCTION} = \frac{\text{OUTPUT RESPONSE}}{\text{INPUT DRIVE}}$$

The transfer function of a system is defined as the complex ratio of the system response to the system forcing function, which, when looking at the frequency response function, is equivalent to forming the ratio of the Fourier transform of the system response to the Fourier transform of the forcing function as follows.

$$H(j\omega) = \frac{\mathcal{F}\{o(t)\}}{\mathcal{F}\{i(t)\}} = \frac{S_y(j\omega)}{S_x(j\omega)}$$

However, from a computational standpoint, this requires the formation of the ratio of two complex quantities. Since this is not a simple computational task, the calculation process can be simplified by rationalizing the denominator of this transfer function expression. So that

$$H(j\omega) = \frac{S_y(j\omega)}{S_x(j\omega)} = \frac{S_y(j\omega) \cdot S_x^*(j\omega)}{S_x(j\omega) \cdot S_x^*(j\omega)}$$

$$= \frac{S_{yx}(j\omega)}{S_{xx}(j\omega)} = \frac{\text{Cross-spectrum}}{\text{Input Power Spectrum}}$$

By rationalizing the denominator, the transfer function computation is reduced to computing the cross spectrum and the input power spectrum.

Ideally, a transfer function measurement is performed on two signals; one representing a unique system forcing function, the other representing the measured system's response. In the absence of any extraneous noise, if no other responses are present and if the system being analyzed is linear, the transfer function measurement should be excellent. however, since the basic cross-spectrum measurement does not give a causality relationship, it has been found useful to define a measurement qualifier. This parameter called the coherence function tries to identify how much of the measured cross-spectrum is related to the measured input and output power spectra. The coherence function, $\gamma^2_{yx}(\omega)$, is defined by

$$\gamma^2_{yx}(\omega) = \frac{|G_{yx}(j\omega)|^2}{G_{xx}(\omega)\, G_{yy}(\omega)} \; ; \; 0 \leq \gamma^2_{yx}(\omega) \leq 1.$$

A coherent output spectrum is formed by taking the product of the coherence function with the spectrum at location A. The result represents the amount of signal level at measurement location A that is coherent with reference signal B. The coherent portion of a total signal is usually measured in terms of a coherent power spectrum which facilitates statistical measurement consideration. Since we are usually interested in measuring the output or response of a system, this has been termed coherent output power. For example, in Figure 6A-3, the magnitude of the Fourier transform of signal A was used, rather than the power spectrum of this signal. In this sense, especially where periodic components are involved, the measurement of total signal level and coherent signal level can be made in terms of the original engineering units, such as sound pressure level in dB, acceleration in g, etc.

$$COP = \frac{\overline{|G_{yx}(j\omega)|^2}}{G_{xx}(\omega) G_{yy}(\omega)} \cdot G_{yy}(\omega) = \frac{\overline{|G_{yx}(j\omega)|^2}}{G_{xx}(\omega)} \cdot \frac{G_{xx}(\omega)}{G_{xx}(\omega)} = \frac{\overline{|G_{yx}(j\omega)|^2}}{\overline{|G_{xx}(\omega)|^2}} \cdot G_{xx}(\omega)$$

$$COP = \overline{|H(j\omega)|^2} \cdot G_{xx}(\omega) .$$

In general, a true coherence function is frequency dependent which can produce a bias error that is also frequency dependent. Commonly used bias, variance and random errors are defined in Table 6A-II. In computation of a coherent output power spectrum, if details of the coherent output power spectrum are important, it is necessary to take a long number of averages. A simple test to see if averaging is sufficient has been developed in which the number of independent ensembles is doubled and the overall level of the coherent output power spectrum observed. If the reduction in the coherent output power spectrum is close to 3 dB for each doubling of the number of independent ensembles, then the bias error is significant and a greater number of averaging samples should be taken. Remember that the coherent output spectrum results solely from the output spectrum and the coherence of the output spectrum with the reference signal, with the level of the reference signal having no effect upon it.

Coherence function and coherent power measurements are useful in determining if the measurement of system transfer functions are valid. When using an FFT processor, the usual technique for computing the transfer function is to form a ratio of the cross-spectrum between system input and output and the system input power spectrum. This produces a "coherent" transfer function measurement as opposed to a transmissibility measurement which is simply the ratio of the system's output power spectrum to its input power spectrum. However, even when a coherent transfer function measurement is made, the coherence function should be observed since it can reveal frequencies or frequency regions where further investigation is warranted. The coherent output power spectrum can then be observed in conjunction with the measured transfer function to isolate measurement regions which might be invalid.

THE EFFECTS OF SIGNAL DELAY ON THE COHERENCE FUNCTION AND COHERENT OUTPUT POWER

When measuring a system transfer function under the condition of broadband excitation into the system, it will sometimes be noted that the coherence function is reduced from unity even though apparently good transfer function information has been developed. A careful evaluation of the different factors and measurement components involved in making the transfer function computation shows that one or more of the following reasons could contribute to this reduction in coherence function and, thus to a reduction in coherent output power.

1. The system being tested might be truly non-linear, especially at resonance.

2. For systems with a high Q, the inherent structural delay through the system at resonance may cause the input and response blocks of data being processed to appear unrelated.

3. Other response signals may be contributing to the system output, as well as the system response related to the true forcing function.

4. System noise near a high Q resonance can be amplified, although it is not part of the actual forcing function.

5. The measured forcing function may tend to 'back off' near a high Q resonance, especially when using electrodynamic exciters, giving a very small forcing function value and a poor cross-spectrum measurement.

6. The response motion measurement at anti-resonance might be very small, once again causing a poor cross-spectrum measurement.

7. Sufficient averaging might not have been performed for the desired measurement to converge.

Delay of the input signal through the transmission media or through a structural system is one of the most common causes of reduction in the coherence function, in structure-borne and airborne noise measurements.

SOURCE IDENTIFICATION AND NOISE REDUCTION

One of the most difficult measurement tasks is to isolate the relative contributions of several potential signal sources from the total measured signal. If the potential sources are independent, often it is possible to use classic cross-correlation techniques in which the measured time delay can indicate possible transmission paths and the peak magnitudes in the cross-correlogram can indicate relative strength. Extreme care must be taken in interpreting correlation functions if periodic functions are present as they can become quite complex. In a measurement of physical parameters, such as noise or vibration, often one will want to pinpoint the primary source of noise contributing to an overall measured sound pressure level. Although the cross-correlation, cross-spectrum and transfer functions measure mutual properties between pairs of signals, none of these functions gives an exact causal relationship. However, the normalized coherence function can in turn show the degree of relationship between a system output and a possible system input. If the coherence function is displayed in conjunction with either the input or output power spectrum or a system transfer function, a determination can be made,using pairs of measurements, as to relative contributions. However, the use of coherent output power measurements permits this same determination of signal contributions in a single, calibrated presentation.

The use of coherent output power measurements in some applications involving noise source identification and noise reduction can reduce the total data processing task almost fifty per cent. The coherence function yields important information concerning causality between analysis locations and can then be used in conjunction with a measured transfer function or measured output power spectrum. The coherent output power offers in a single measurement the amount of power relating to a given reference source. If uncorrelated sources are involved, the total measured output power spectrum will be the sum of the individual coherent output power spectra. If related noise sources are involved, the concepts of partial coherence, a tool still in the development stages, must be employed together with the resulting coherent power spectrum measurements.

TRANSMISSIBILITY MEASUREMENTS

Transfer function measurements performed using the FFT approach are typically made through computations involving the cross-spectrum between acceleration and foce and the power spectrum of the applied force. Although the cross-spectrum and, therefore, the transfer function, does not give an exact causality relationship, the cross-spectrum does tend to reject components which are not usually present in both signal inputs. For this reason, the transfer function as calculated through the ratio of the cross-spectrum to the input power spectrum is sometimes termed a "coherent" transfer function measurement.

In some applications, however, it may be necessary to measure a total transfer of energy regardless of any coherent sources or effects which may be present. This total transfer ratio is often termed transmissibility. The standard definition of transmissibility as calculated by the SD360 is

$$T = \left[\frac{G_{yy}(\omega)}{G_{xx}(\omega)}\right]^{1/2} = |H(j\omega)|$$

$$\text{Since } \frac{G_{yy}(\omega)}{G_{xx}(\omega)} = |H(j\omega)|^2$$

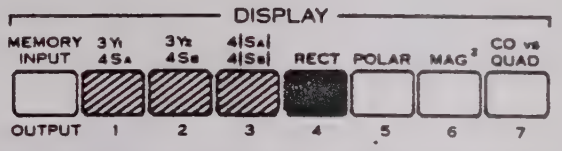
The ratio of a system's output to input power spectrum produces a term proportional to the square of the magnitude of the system frequency response function. If the square root of this term is taken, the frequency response function magnitude is obtained and this will be identical to the magnitude of the transfer function as measured by our standard transfer function technique provided that no incoherent signals are present. The transmissibility function is usually a dimensionless ratio. Transmissibility ratios are often measured on structures using two accelerometers placed at different points of interest on the test structure. Transmissibility functions are also used to measure the transmission of sound between two points and determine sound attenuation or reduction factors as a function of material properties, geometry, etc.

OPERATING INSTRUCTIONS FOR COHERENCE FUNCTION

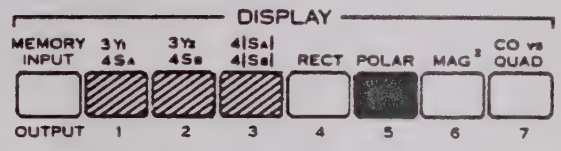
■ = IN □ = OUT ▨ = CHOICE

Function 2

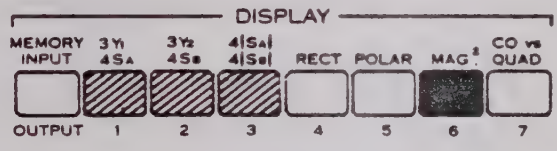
Rectangular Coordinates
(a + jb) (1, 2, 3 or none)



Polar Coordinates
($|S_X|$, $<\theta$) (1, 2, 3 or none)



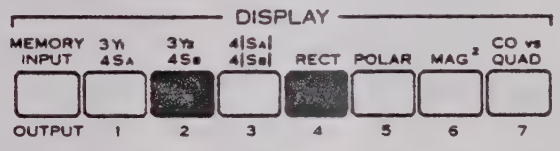
Magnitude² Coordinates
($|S_X|^2$, $<\theta$) (1, 2, 3 or none)



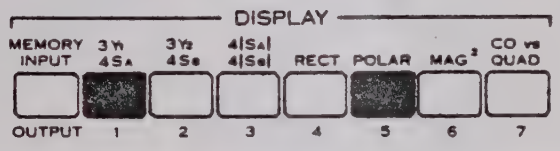
Function 3

Real-Rectangular Coordinates

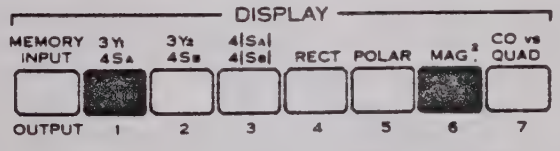
Imaginary-Rectangular Coordinates



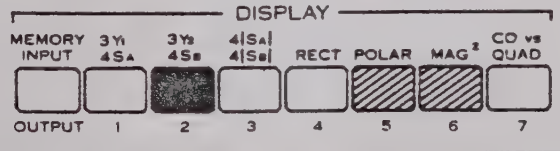
Magnitude



Magnitude²



Phase
(Either Button 5 or Button 6 may be selected.)



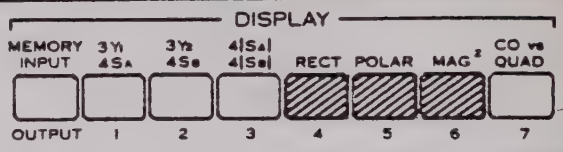
OPERATING INSTRUCTIONS FOR COHERENCE FUNCTION

Function 6

Cross-spectrum (G_{yx})

	Y_1	Y_2
4	REAL	IMAG
5	MAG	PHASE
6	MAG^2	PHASE

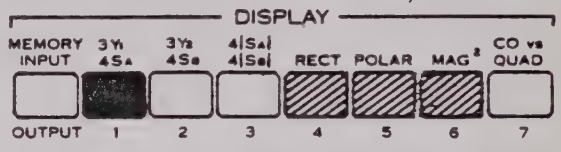
COP: T.F.



Transfer Function and Phase

	Y_1	Y_2
4	REAL	IMAG
5	MAG	PHASE
6	MAG^2	PHASE

COP: T.F.

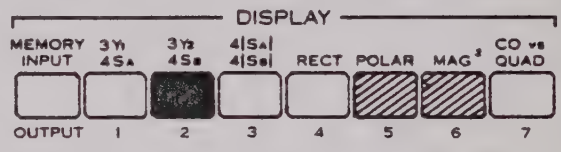


Transfer Function and Coherence

Set q switch to 7

	Y_1	Y_2
5	MAG	γ_{yx}^2
6	MAG^2	γ_{yx}^2

COP: T.F.



Coherent Output Power and Phase

Y_1	Y_2
MAG^2	PHASE

COP: T.F.

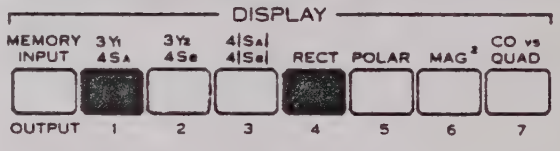
Power Spectra (G_{xx} and G_{yy})

	Y_1	Y_2
5	MAG	MAG
6	MAG^2	MAG^2

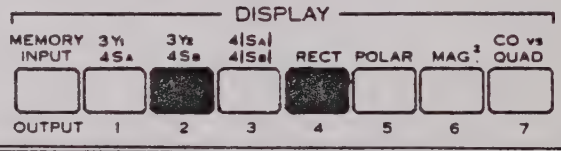
COP: T.F.

Function 7

EVEN Array



ODD Array



Function 8

Correct Real Display (Hermitian Symmetry)

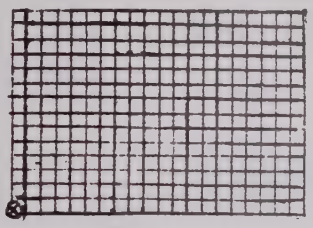
Function 6 TRANSFER FUNCTION AND PHASE

1 LIN X
FREQUENCY

CALIBRATE
ON LOW HIGH
OFF

.0

Hz deg lin
ms dB log

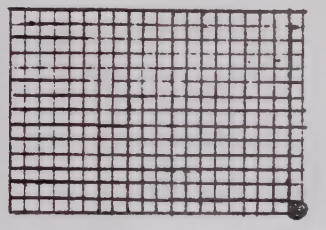


2 LIN X
FREQUENCY

CALIBRATE
ON LOW HIGH
OFF

5000.0

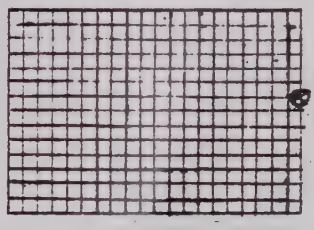
Hz deg lin
ms dB log



3 LOG Y_1
TRANSFER FUNCTION
MAGNITUDE

CALIBRATE
ON LOW HIGH
OFF

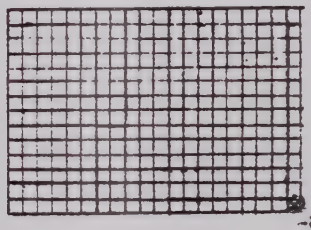
Hz deg lin
ms dB log



4 LOG Y_1
TRANSFER FUNCTION
MAGNITUDE

CALIBRATE
ON LOW HIGH
OFF

Hz deg lin
ms dB log

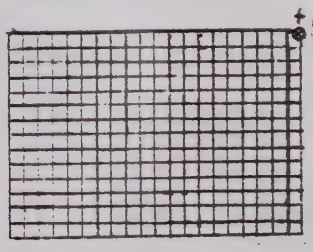


5 LIN Y_2
PHASE

CALIBRATE
ON LOW HIGH
OFF

179.9

Hz deg lin
ms dB log

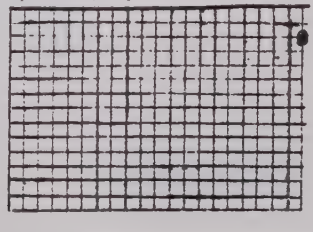


6 LIN Y_2
PHASE

CALIBRATE
ON LOW HIGH
OFF

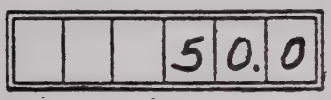
.0

Hz deg lin
ms dB log



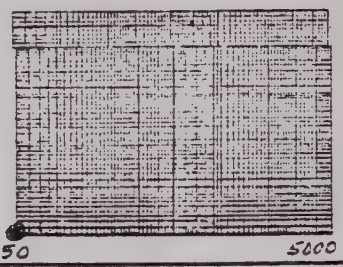
Function 6 TRANSFER FUNCTION AND COHERENCE

7 LOG X FREQUENCY

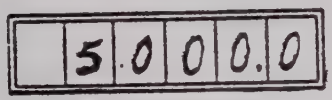


CALIBRATE — ON LOW HIGH OFF

Hz deg lin / ms dB log

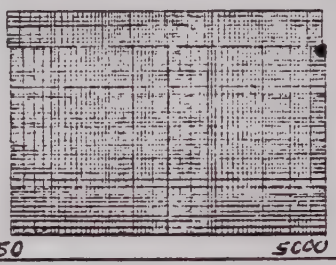


8 LOG X FREQUENCY



CALIBRATE — ON LOW HIGH OFF

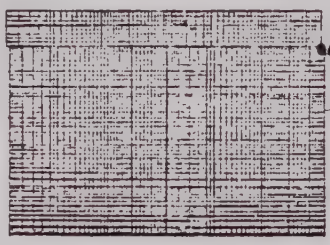
Hz deg lin / ms dB log



9 LOG Y_1 TRANSFER FUNCTION MAGNITUDE

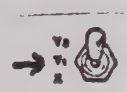
CALIBRATE — ON LOW HIGH OFF

Hz deg lin / ms dB log

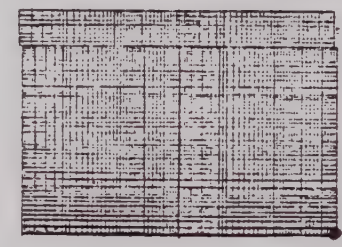


10 LOG Y_1 TRANSFER FUNCTION MAGNITUDE

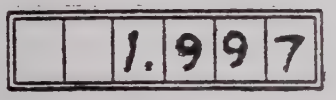
CALIBRATE — ON LOW HIGH OFF



Hz deg lin / ms dB log

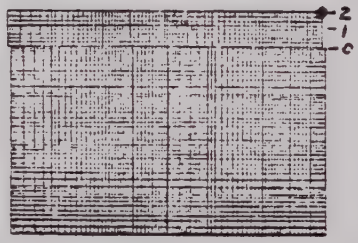


11 LIN Y_2 COHERENCE

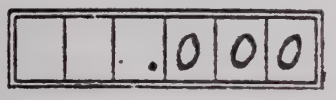


CALIBRATE — ON LOW HIGH OFF

Hz deg lin / ms dB log

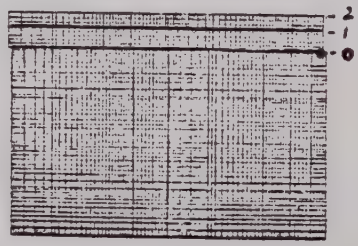


12 LIN Y_2 COHERENCE



CALIBRATE — ON LOW HIGH OFF

Hz deg lin / ms dB log



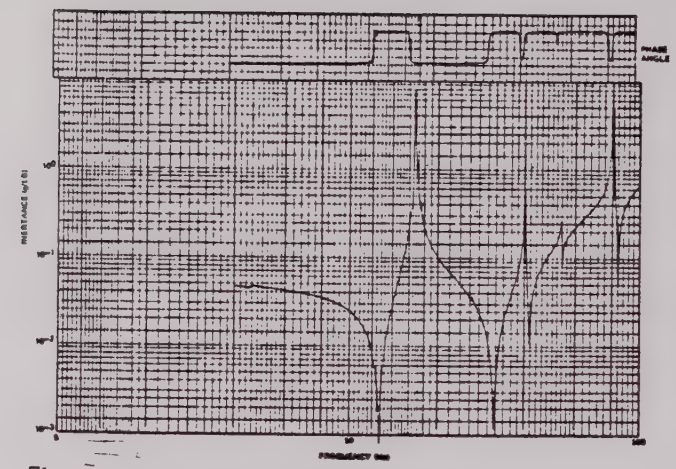


Figure 6A-1. Inertance Response (20-500 Hz)

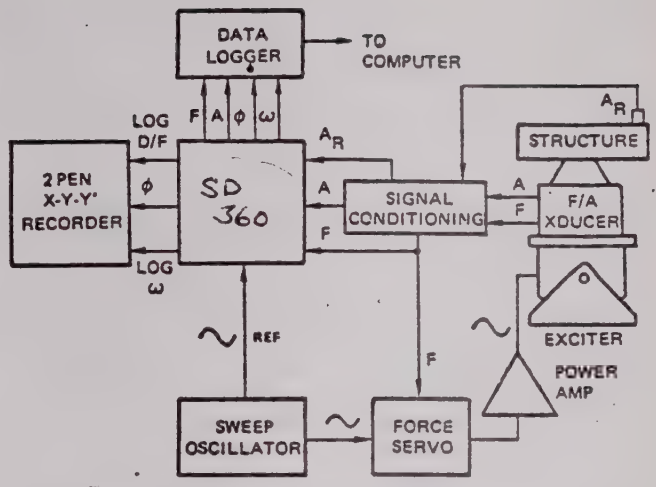


Figure 6A-2. Instrumentation for Structural Transfer Function Measurements

TABLE 6A-II

DEFINITION OF COHERENCE FUNCTION ERROR

For $\gamma^2_{yx}(\omega) = 0$ and $n_d \geq 32$

where n_d is number of disjoint time records

$$\text{Bias}\,[\gamma^2_{yx}(\omega)] \approx \frac{1}{n_d}$$

$$\text{Variance}\,[\gamma^2_{yx}(\omega)] \approx \frac{1}{n_d^2}$$

For $0 \leq \gamma^2_{yx}(\omega) \leq 1$ and $n_d \geq 32$

$$\text{Bias}\,[\gamma^2_{yx}(\omega)] \approx \frac{1}{n_d}(1 - \gamma^2_{yx}(\omega))^2$$

$$\text{Variance}\,[\gamma^2_{yx}(\omega)] \approx \frac{2\gamma^2_{yx}(\omega)}{n_d}(1 - \gamma^2_{yx}(\omega))^2$$

For $\gamma^2_{yx}(\omega) \geq 0$ and $n_d \geq 32$ the normalized bias error and normalized random error can be given as

$$\epsilon_B\,[\gamma^2_{yx}(\omega)] = \frac{\text{Bias}\,[\gamma^2_{yx}(\omega)]}{\gamma^2_{yx}(\omega)} \approx \frac{[1 - \gamma^2_{yx}(\omega)]^2}{n_d\,\gamma^2_{yx}(\omega)}$$

$$\epsilon_R\,[\gamma^2_{yx}(\omega)] = \frac{\sqrt{\text{Variance}\,[\gamma^2_{yx}(\omega)]}}{\gamma^2_{yx}(\omega)} \approx \frac{\sqrt{2}\,[1 - \gamma^2_{yx}(\omega)]}{n_d\,[\gamma_{yx}(\omega)]}$$

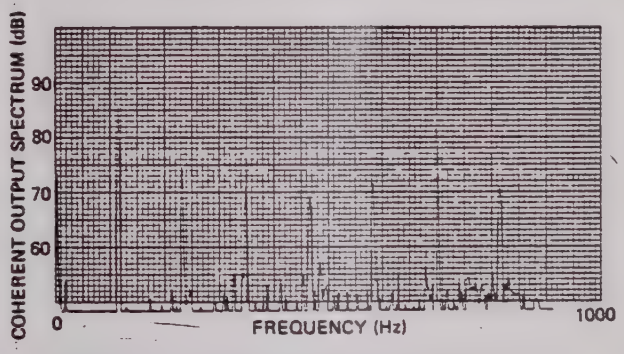


Figure 6A-3. Coherent Output Spectrum Location y Reference-Location x

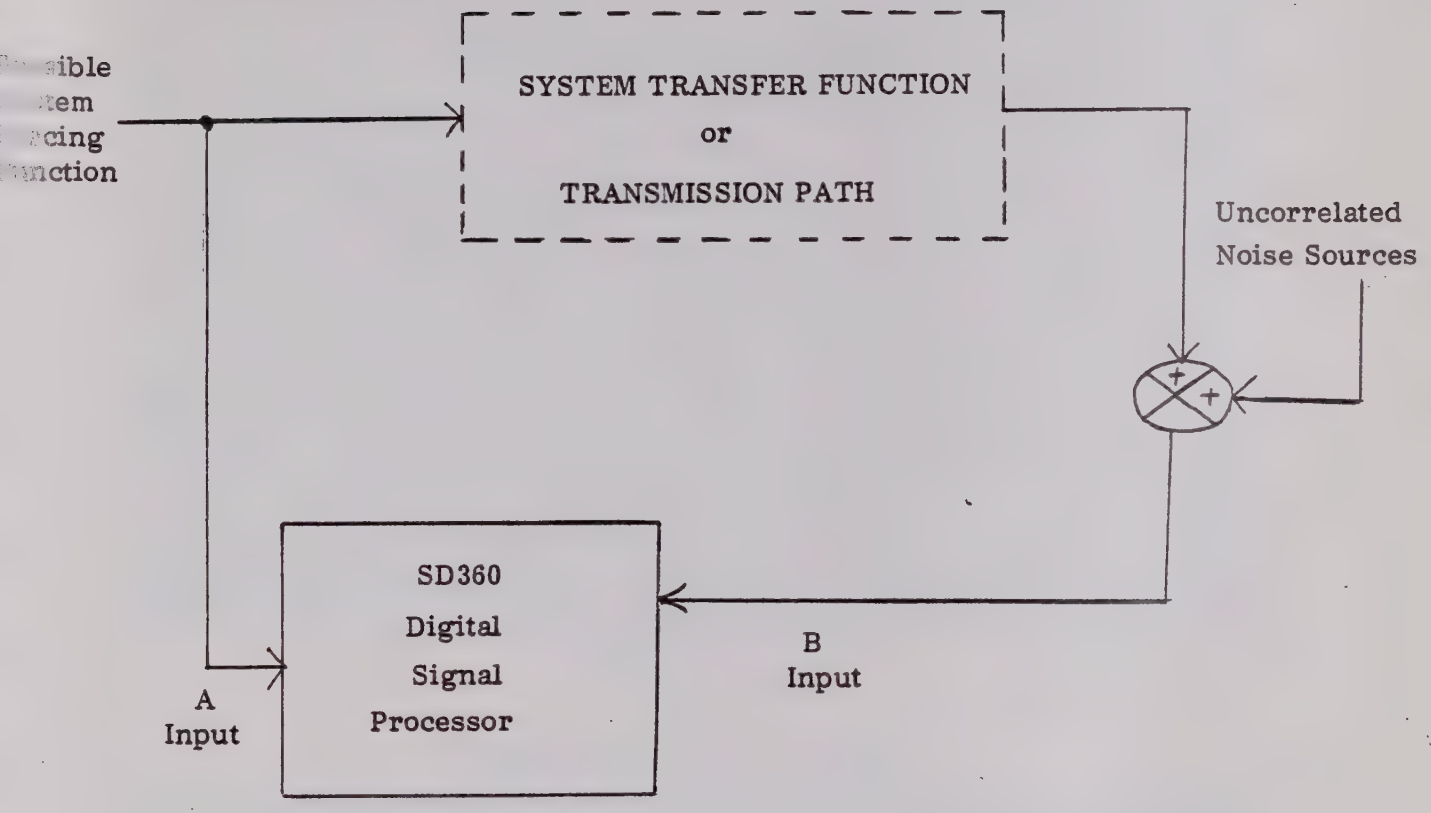


Figure 43. Measuring Coherent Output Power

STEP 1. — Measure G_{BB}, the total power Spectrum of B

STEP 2. — Measure $\gamma^2 G_{BB}$, the Coherent Output Power — the amount of power in B due to the reference (A) input.

TYPICAL STRUCTURAL TRANSFER
FUNCTION ANALYSIS

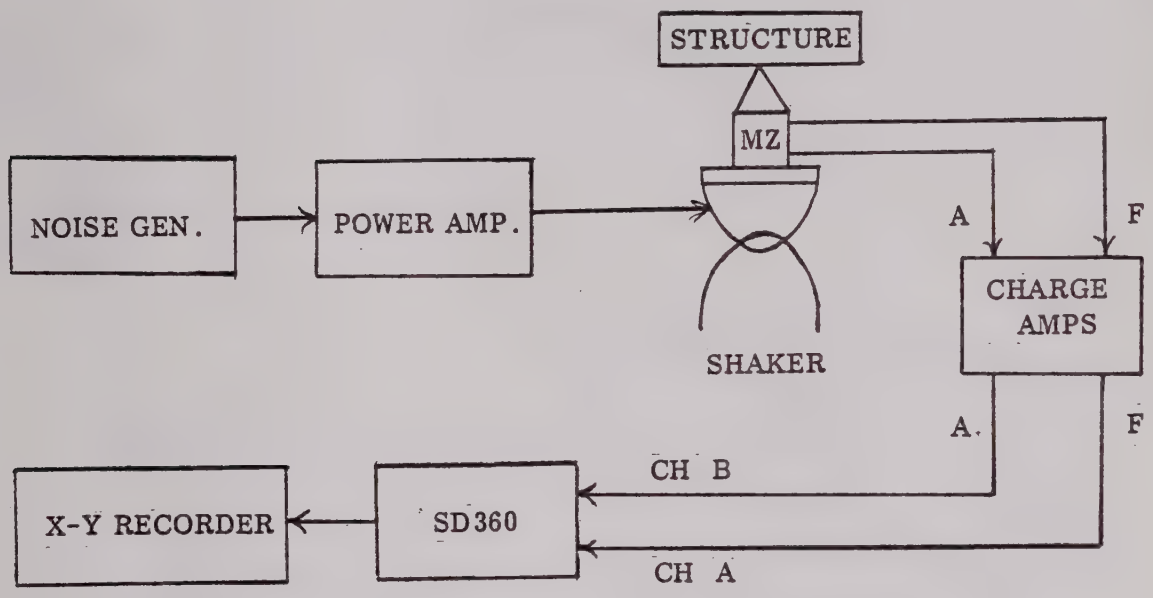


Figure 44. Typical Structral Transfer Function Analysis

STEP 1 — Using random excitation perform a Function 4 — 2-channel PSD on Force (F) and Acceleration (A). (Figure 45).

STEP 2 — (Optional) Perform a Function 5 — Cross Spectrum between F and A. (Figure 46).

STEP 3 — Perform a Function 6 — Transfer Function A/F — Magnitude and Phase. (Figure 47).

STEP 4 — Display Function 6 in real and imaginary components. (Figure 48).

Figure 45. Function 4 — Power Spectral Density

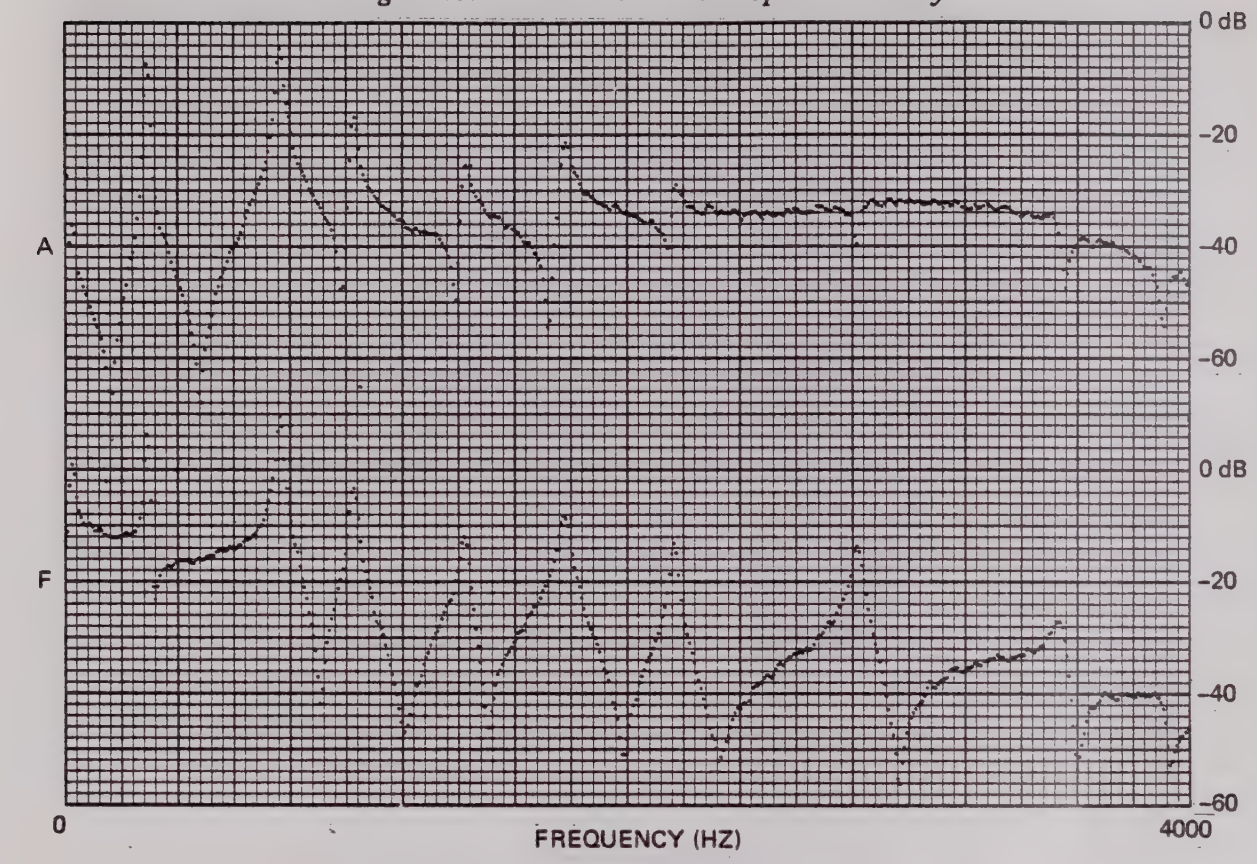


Figure 46. Function 5 — Cross-Spectrum

Figure 47. Function 6 — Transfer Function

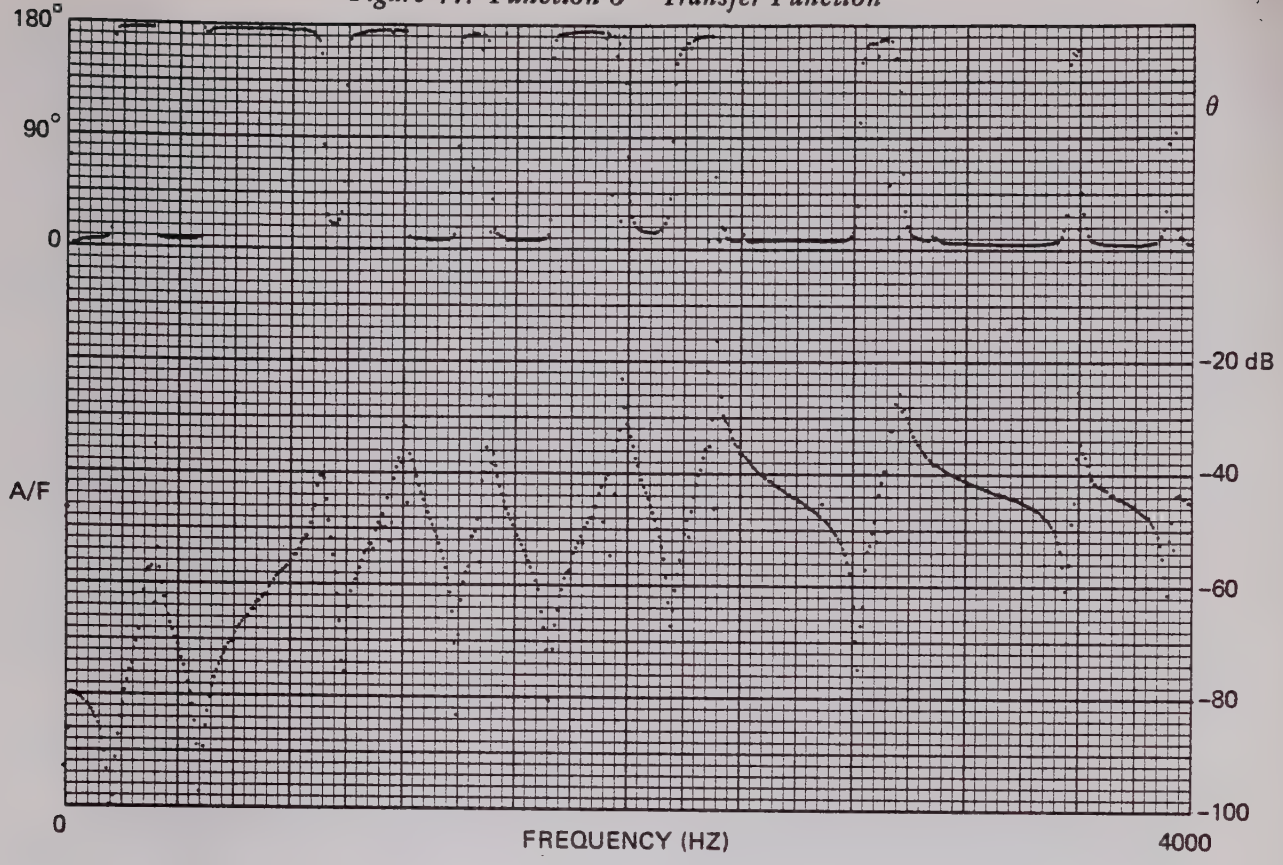
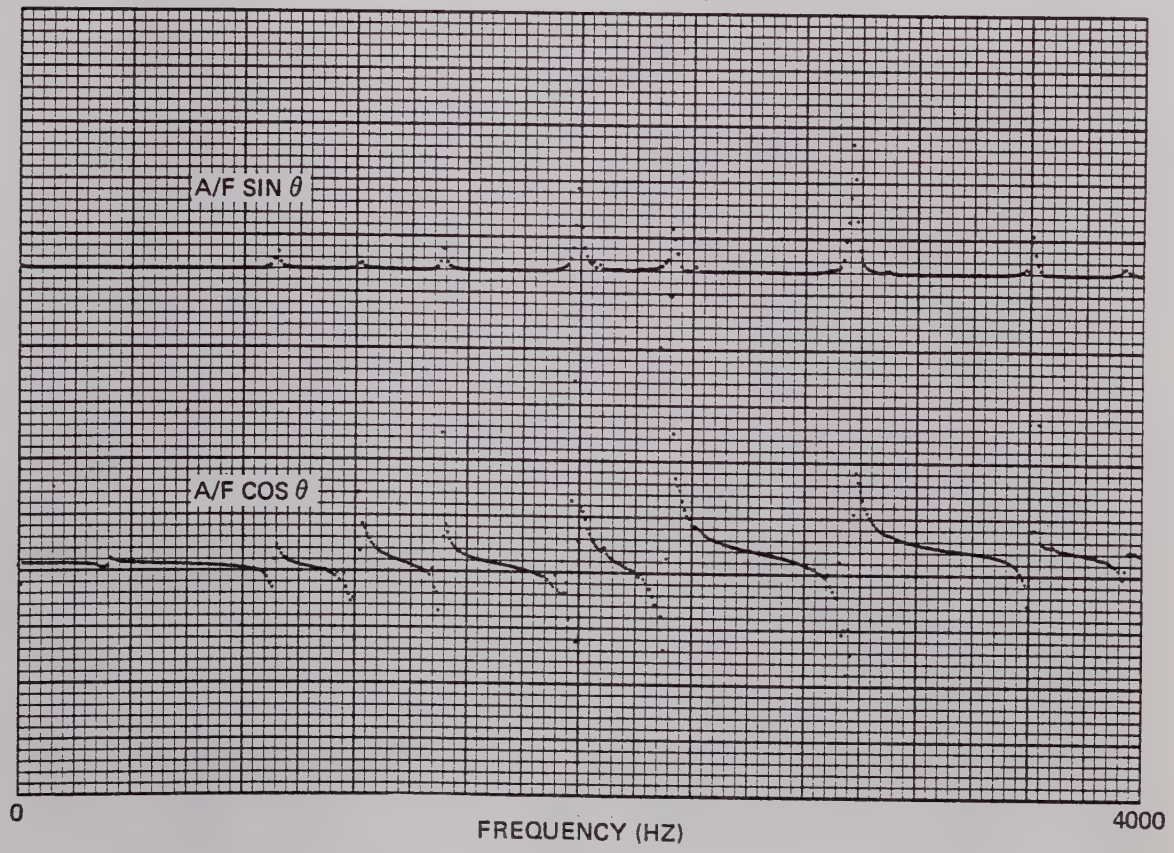


Figure 48. Function 6 — Transfer Function



TRANSFER FUNCTION SEQUENCE
ELECTRICAL NETWORK TEST

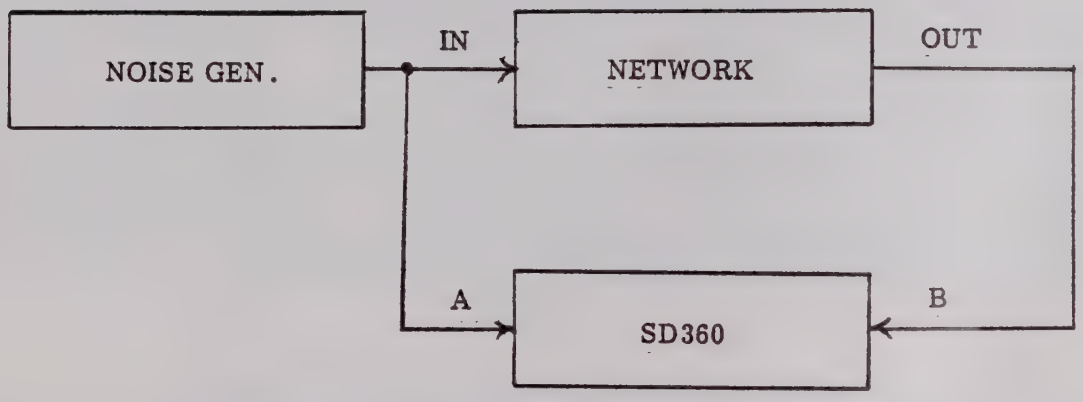


Figure 49. Transfer Function Sequence Electrical Network Test

STEP 1 — Input and Output Raw Data Waveforms (Figure 50).

STEP 2 — Perform a Function 4 with 64 averages of transforms of input (cause) and output (effect) (Figure 51).

STEP 3 — Perform a Function 6 and display as magnitude and phase of transfer function B/A. (Figure 52).

STEP 4 — Display Function 6 as the real and Imaginary part of the transfer function; B/A cos θ and B/A sin θ. (Figure 53).

STEP 5 — Display Function 6 as the Co vs Quad (Nyquist Diagram) of the transfer function. (Figure 54).

Figure 50. Raw Data Waveforms

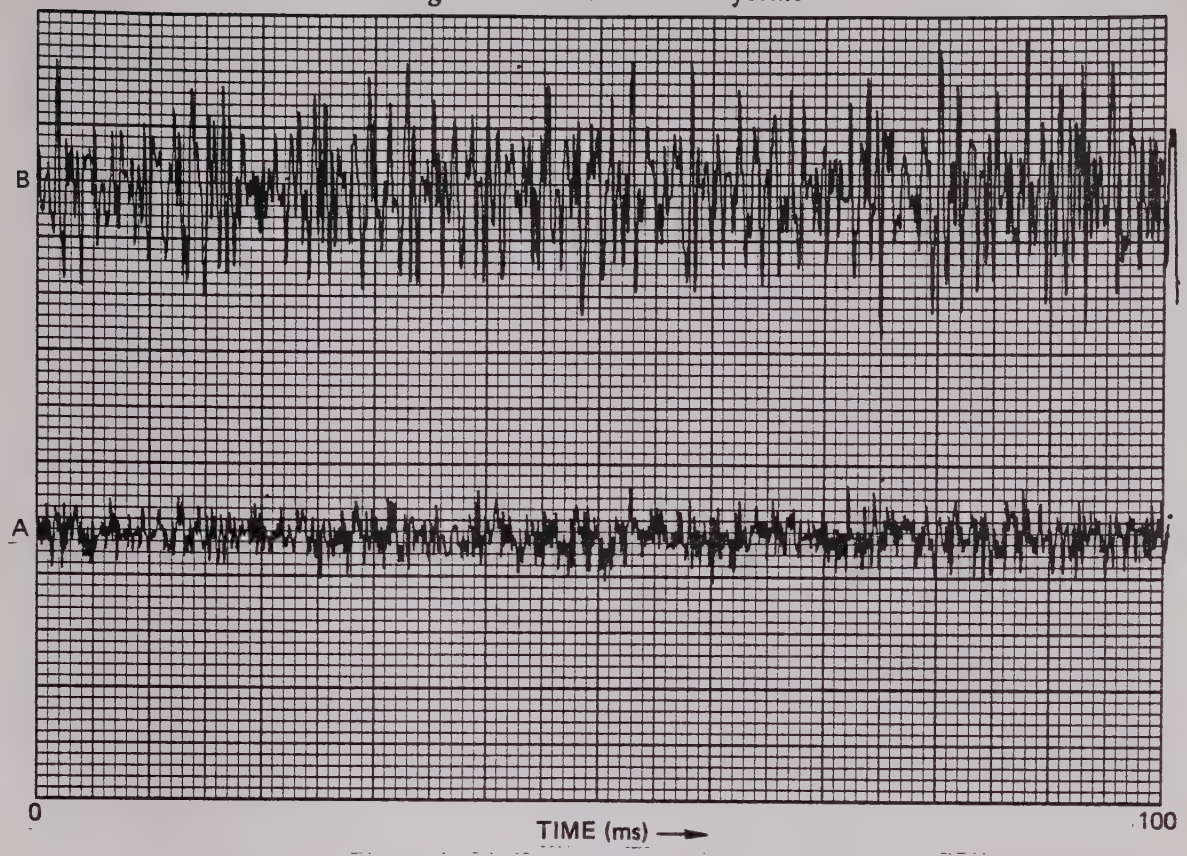


Figure 51. Function 4 – Dual Fourier Transform

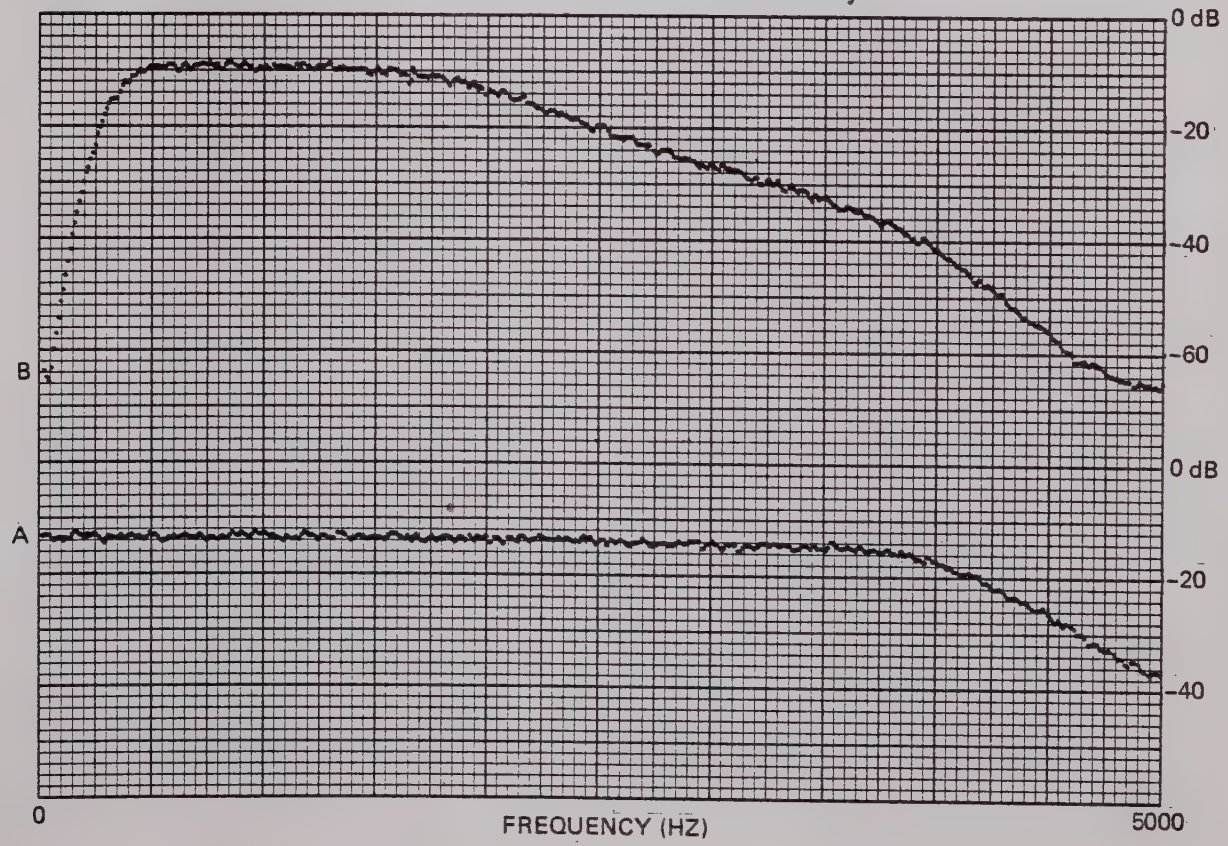


Figure 52. Function 6 — Transfer Function-Polar

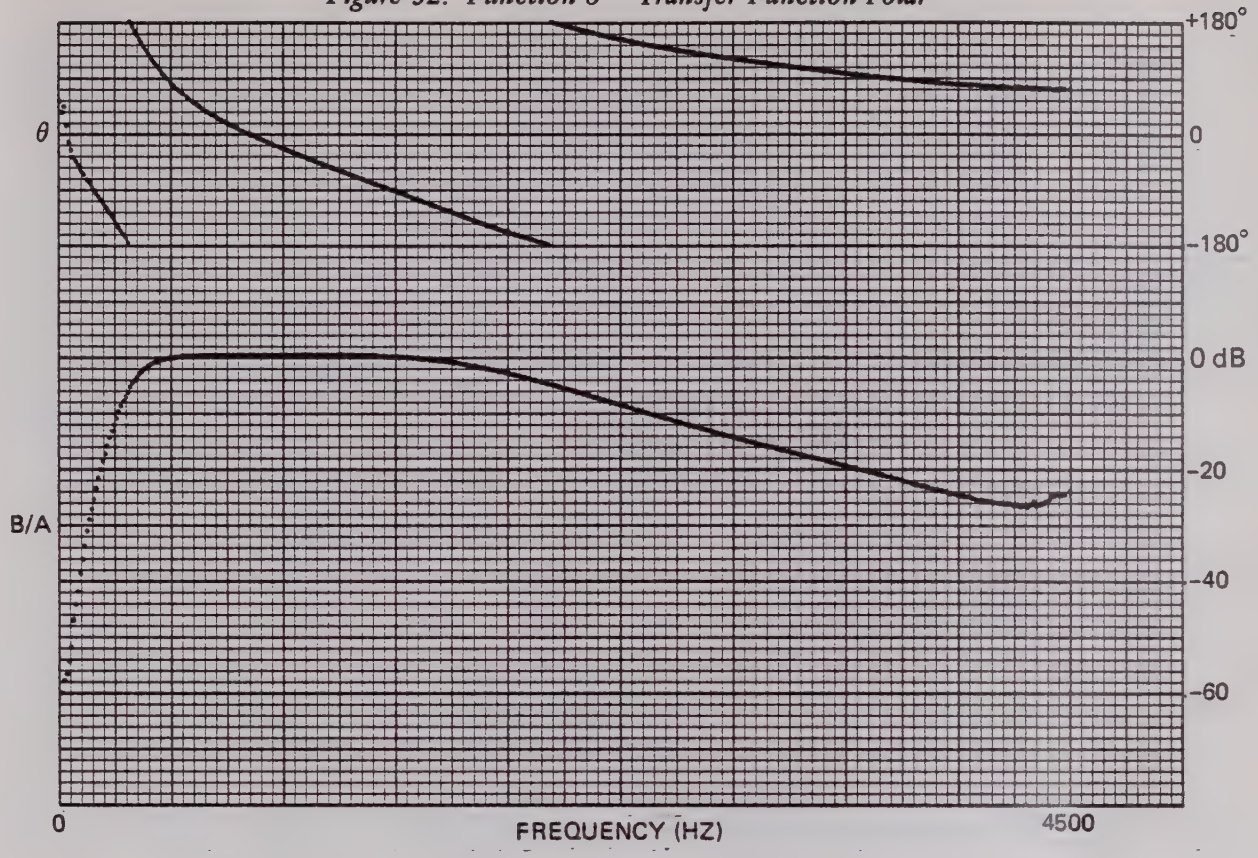


Figure 53. Function 6 — Transfer Function-Rectangular

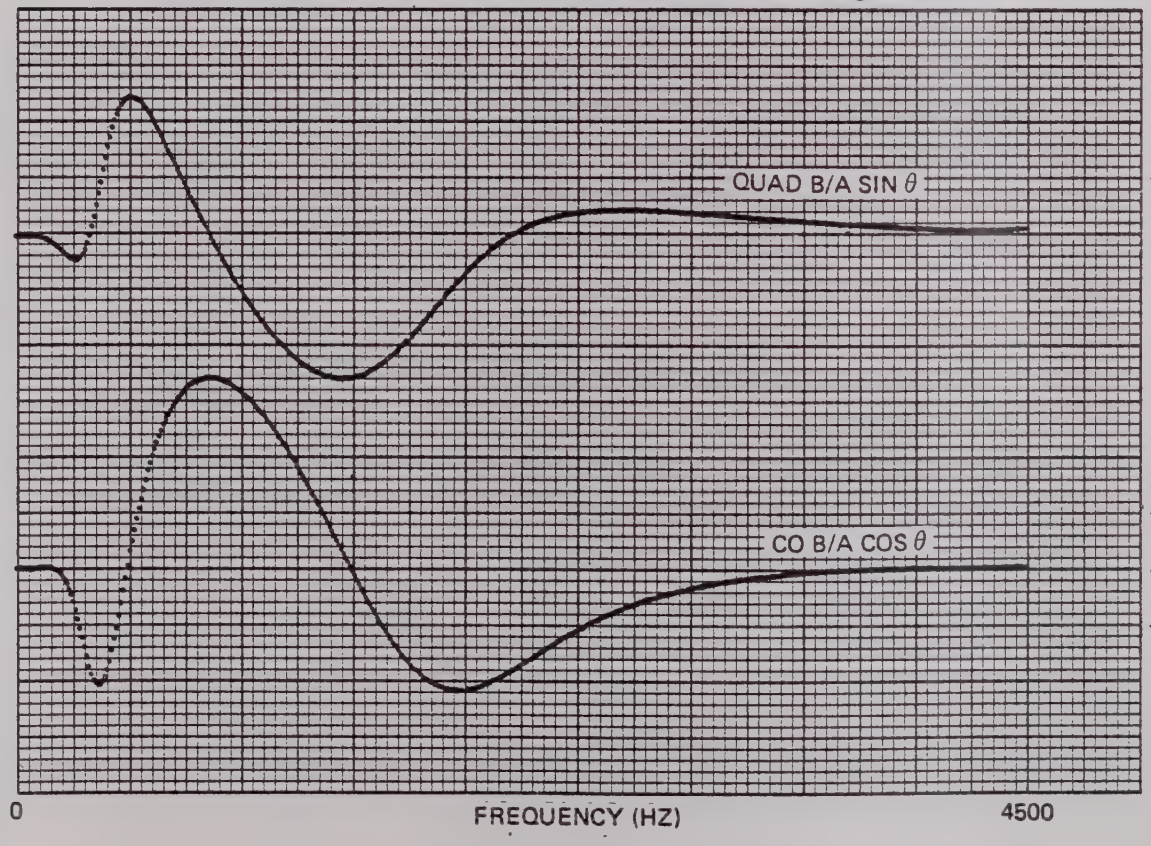
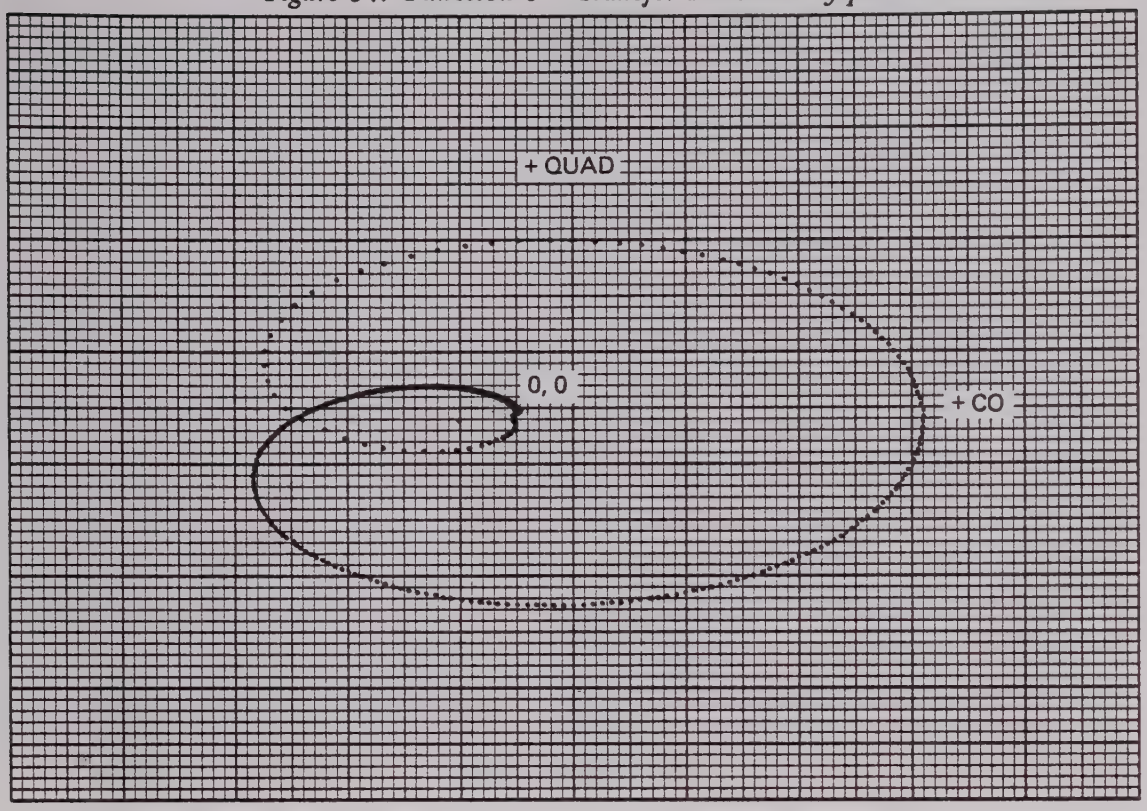


Figure 54. Function 6 – Transfer Function-Nyquist



FUNCTION 7 – INVERSE TRANSFORM OF THE INPUT MEMORY

Function 7 performs an inverse Fourier transform of frequency domain data that is loaded in MEMORY A or B, or MEMORY A and B. The data is usually entered into the input buffer by means of the EDIT MODIFY Function and the CURSOR joystick. Rectangular coordinates are chosen for the display. A given sequence in selecting the display controls is required to produce a continuous time function in the output memory as a result of the inverse transform.

1. Using the Function 7 diagram, select the front panel controls designated ■ , and set the thumbwheel switches to the number designated by Σ . The MEMORY HOLD A & B light should be on.

2. Momentarily, press Cursor RESET button and deflect the Cursor joystick to the right. This will zero out both input memories.

3. Release EDIT INPUT B and select the EDIT MODIFY function. Use the Cursor control to draw in the desired frequency spectrum information.

4. Release the EDIT INPUT ON and the DISPLAY MEMORY INPUT buttons.

5. Momentarily push the controls designated ⊠ , and push EXECUTE ⊗ .

6. Adjust the q switch for a maximum display amplitude without output overload.

7. Press the MEMORY HOLD OUTPUT button.

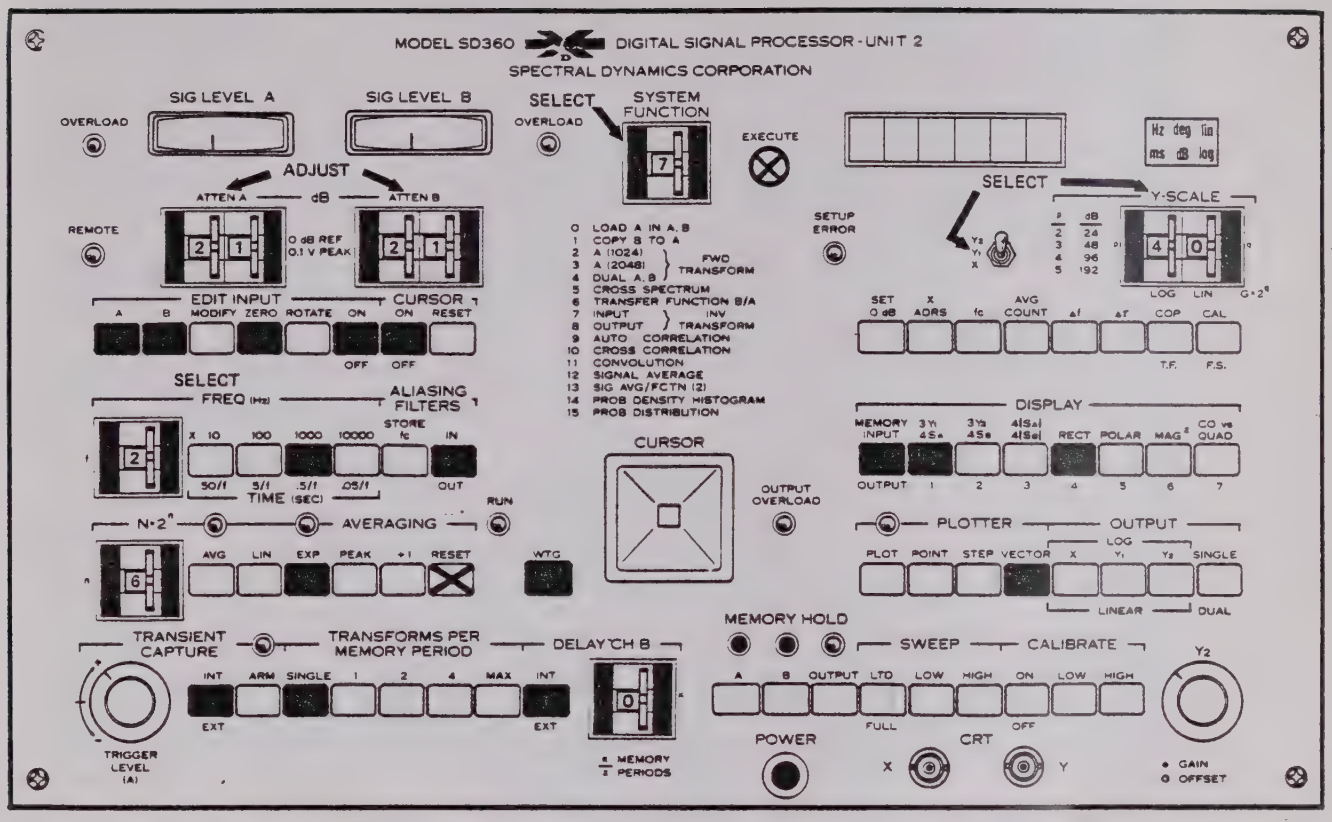


Figure 55. Function 7 – Inverse Transform Of The Input Memory Diagram

8. Select System Function 9 on the thumbwheel switch and press the EXECUTE button once. The output display now represents all 1024 data points on a single trace. The execution of a Function 9 in this case has the effect of swapping around the position of the data in the Output Memory to produce the continuous time waveform observed on the display.

PERIODIC SIGNAL SPECTRUM SIMULATION

Another example of the Inverse Transform operation of the SD360 is periodic signal spectrum simulation in the input memory.

1. Using the Function 7 diagram select the front panel controls designated ■. The Memory Hold A and B lights should be on.

2. Momentarily, press the CURSOR RESET button and deflect the CURSOR control to the right to zero both input memories.

3. Press the X-ADRX (X-axis address) push button and place the axis toggle switch to the X position.

4. Release the EDIT INPUT ON, A and B push buttons and push CURSOR RESET.

A random number between 0 and 1023 will appear in the LED display, since the MEMORY HOLD A control had been pushed at some random time and the X-address counter was stopped at this random address. Since CURSOR RESET was pushed, the X-address showing in the LED is effectively X-address 0.

For example: Assume the X-address reading in the LED display is 653 and we wish to write in memory location 5. Move the CURSOR control until the LED reads 658, which is effectively X-address location 0005.

The drawing of equivalent spectrum lines into input memory assigns the INPUT MEMORY A the function of the real array and INPUT MEMORY B, the function of the imaginary array in an equivalent Fourier transform. The positive real component drawn into input memory corresponds to both a sine and a cosine signal. A Fourier transform of a sine wave has a positive real part and a negative imaginary part. Conversely, the Fourier transform of a cosine wave has a positive real part and a positive imaginary part. By drawing in a positive real component (MEMORY A) and a zero imaginary component (MEMORY B) a signal can be created which has both sine and cosine terms.

5. Select EDIT INPUT ON, A and MODIFY controls. Deflect the CURSOR Control upwards and write in a full scale single positive spectrum line in X-address location 5.

6. Release the EDIT INPUT ON control.

7. Release the DISPLAY MEMORY INPUT button and push the AVERAGING RESET and EXECUTE controls.

8. Adjust the q switch until 5 full periods of a cosine wave (even function) are clearly displayed.

9. Select DISPLAY $3Y_2 4S_B$ control and momentarily push the EXECUTE button. Five full periods of a sine wave (odd function) will appear in the display.

10. Select the DISPLAY MEMORY input control and move the CURSOR to X-address location 668, which corresponds to X-address 15 of the third harmonic of our perfect spectrum line.

11. Select the EDIT INPUT ON control and deflect the CURSOR in the vertical direction. An inverse transform of these two positive input spectrum components will yield a time domain waveform consisting of the fundamental and third harmonic.

12. This can be observed by releasing the DISPLAY MEMORY INPUT control and pushing the EXECUTE button. The display presented will be an Odd function. The Even function can be displayed by selecting DISPLAY $3Y_1 4S_A$ button and pushing the EXECUTE button.

Each display presents a very continuous-looking periodic signal which consists of a full 1024 points in each case. It can be seen that the number of cycles produced in a waveform is directly proportional to the X-address number of the synthesized spectrum component.

13. Continuing on with the above example, the phase angle between the fundamental and third harmonic can be changed by introducing an imaginary component in Channel B Input Memory. Release the Edit INPUT MEMORY A and select MEMORY B. The X-address of the CURSOR should still be set at 15.

14. Select the EDIT INPUT MODIFY and ON controls. Deflect the CURSOR control in the vertical direction, introducing a positive imaginary component.

15. Release the EDIT INPUT ON and the DISPLAY MEMORY INPUT controls. Push the EXECUTE button and observe the contribution of the imaginary term to the overall time domain waveform. By performing an Inverse Transform on real and imaginary components that are written into the Input Memories on the SD360 using the EDIT feature, almost any time domain waveform can be synthesized.

Figure 56. Input Memory – Real Array

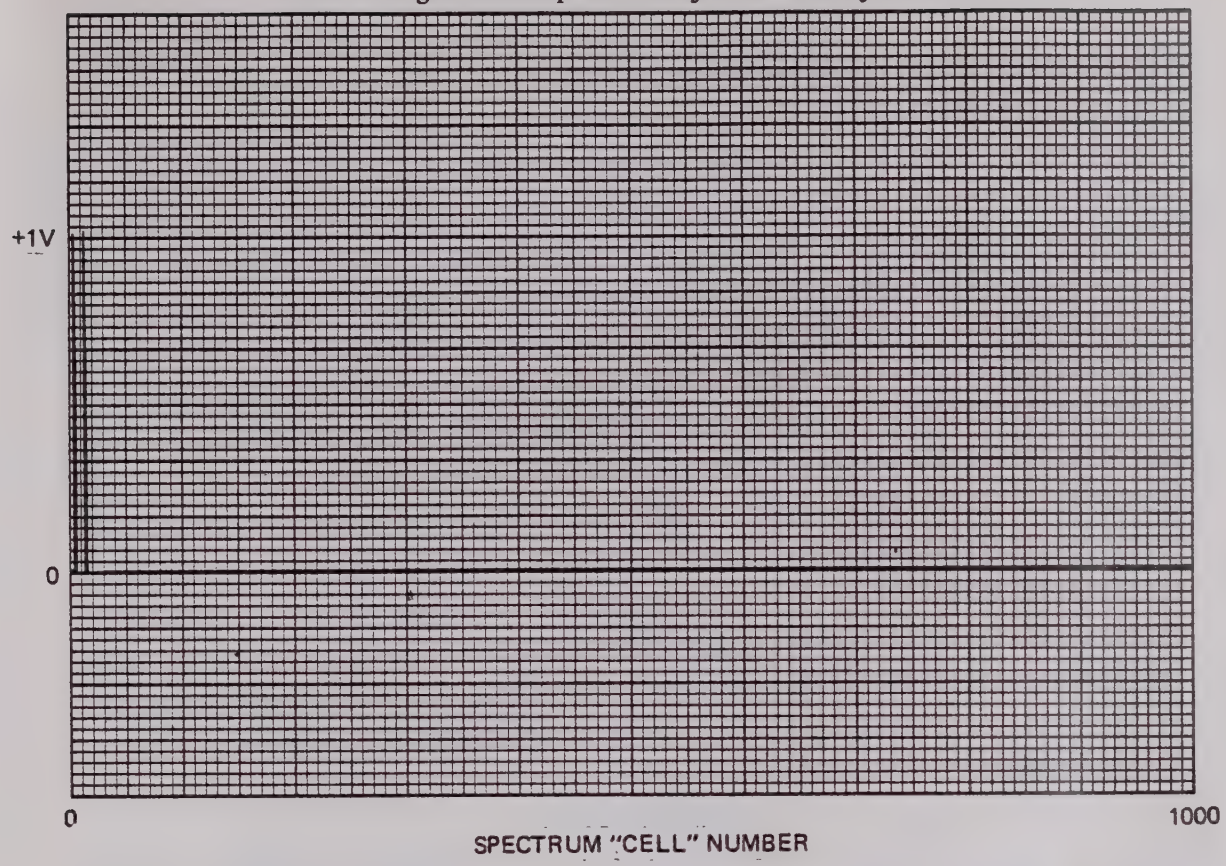
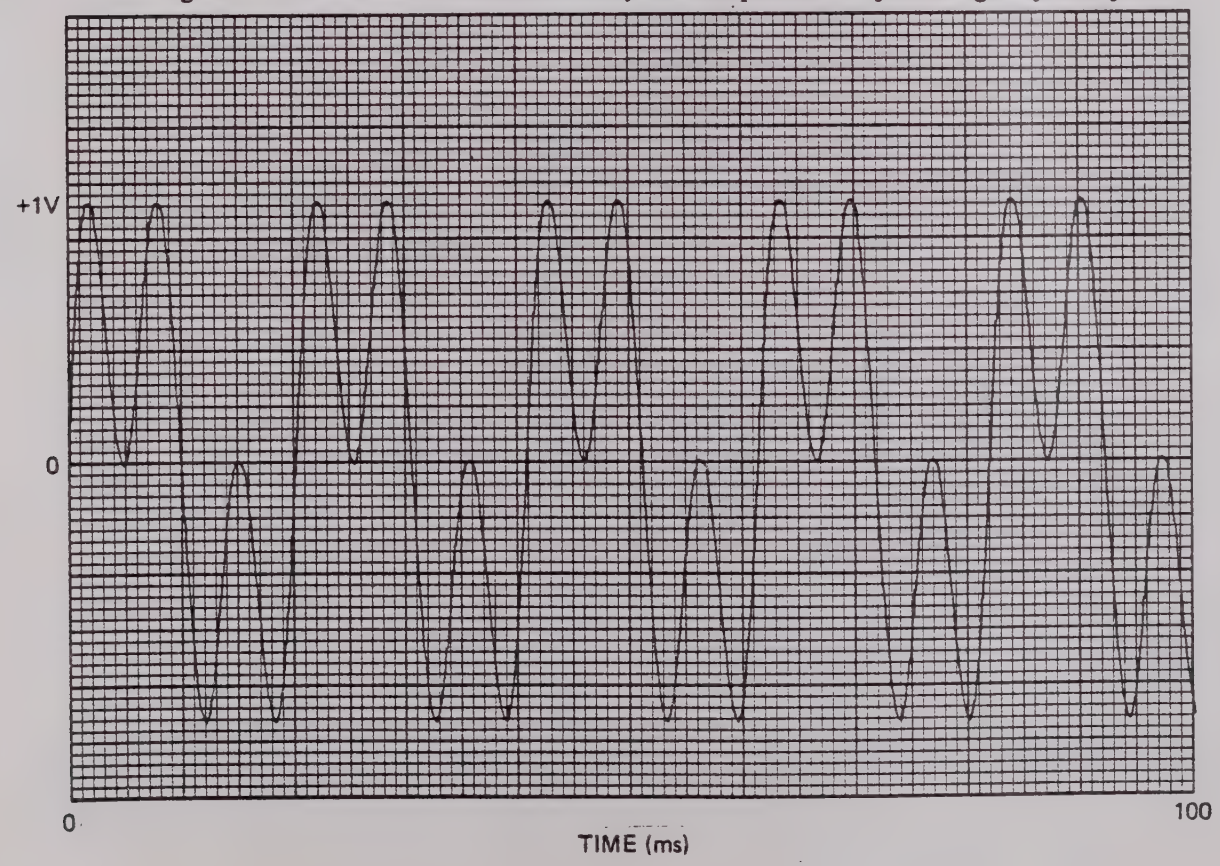


Figure 57. Function 7 – Inverse Transform Output Memory – Imaginary Array



FUNCTION 8 – INVERSE TRANSFORM OF THE OUTPUT MEMORY

Data that is stored in the Output Memory in rectangular coordinates as a result of performing a Fourier Transform using Functions 2 or 4 can be inverse Fourier transformed using Function 8. In several applications, it might be desirable to look backward into the time domain and see what type of time history existed to produce the resultant Fourier transform. Data that has been processed using Function 5 can also be inverse Fourier transformed from the Output Memory. Function 8 requires that the data in the Output Memory be stored in rectangular coordinates. The SD360 cannot average data in rectangular coordinates, so the Inverse Transform of contents in the Output Memory is performed on unaveraged data. The only exception is in Function 5 where the SD360 can average two signals in rectangular coordinates.

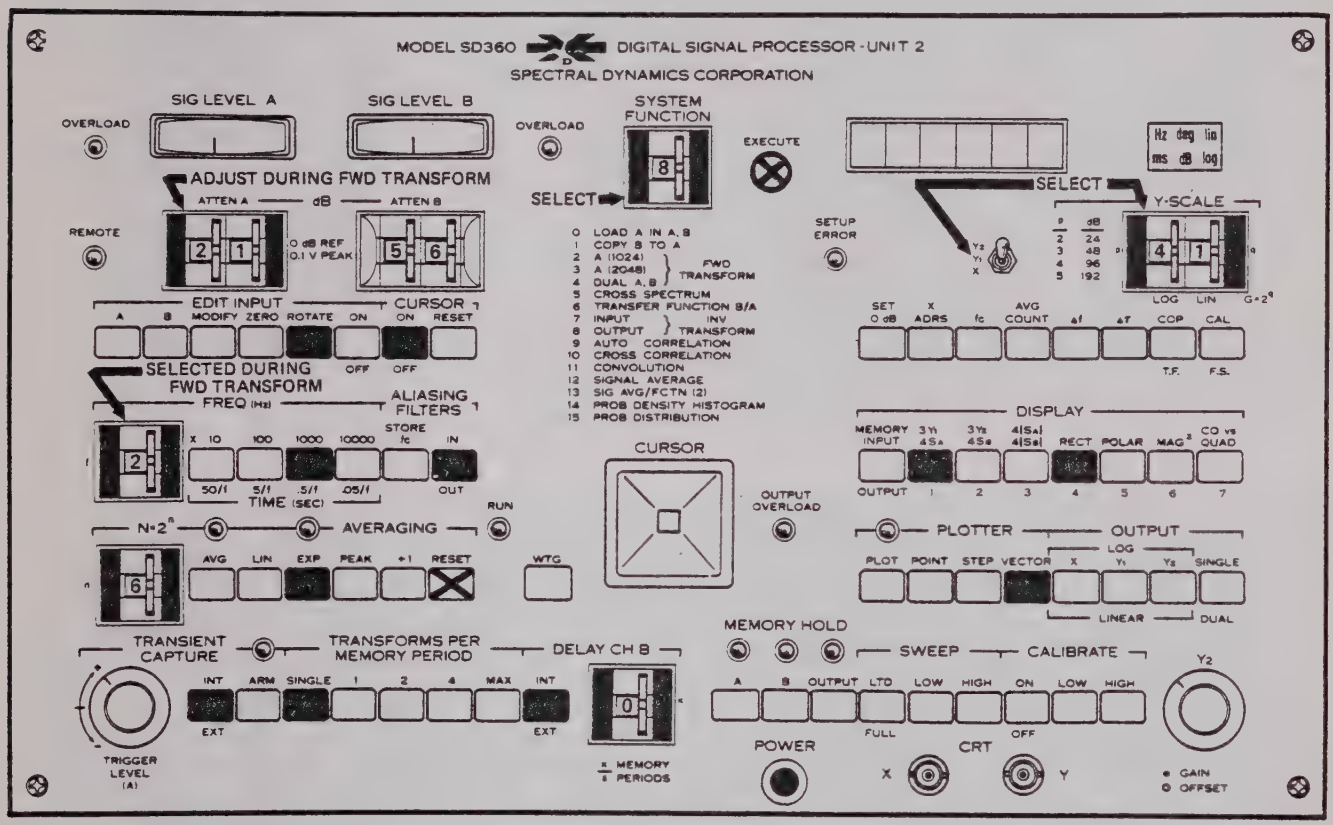


Figure 58. Function 8 Diagram

1. The performance of a Function 8 presumes that either a Function 2 or 4 has been performed on the data of interest without averaging and the results displayed in rectangular coordinates. With this condition satisfied, using the Function 8 diagram, select the front panel controls designated , and set the thumbwheel switches to the number designated by . The frequency range and input attenuation settings will have been established when the Fourier Transform was performed.

2. Momentarily, press AVERAGING RESET and the EXECUTE buttons.

3. Adjust the q switch until a desirable signal amplitude is observed on the display. Figures 59 and 60 are examples of a Function 2 and a Function 8.

The above procedure was performed using rectangular weighting. If a symmetrical weighting function is used (with the WTG button in) on the original data that was processed with a Fourier Transform and stored in the Output Memory in rectangular coordinates, the shape of the symmetrical weighting function would be observed in the shape of the time domain signal resulting from the Inverse Transform of the Output Memory (Figures 61 and 62).

Figure 59. Function 2 – Forward Transform-Rectangular

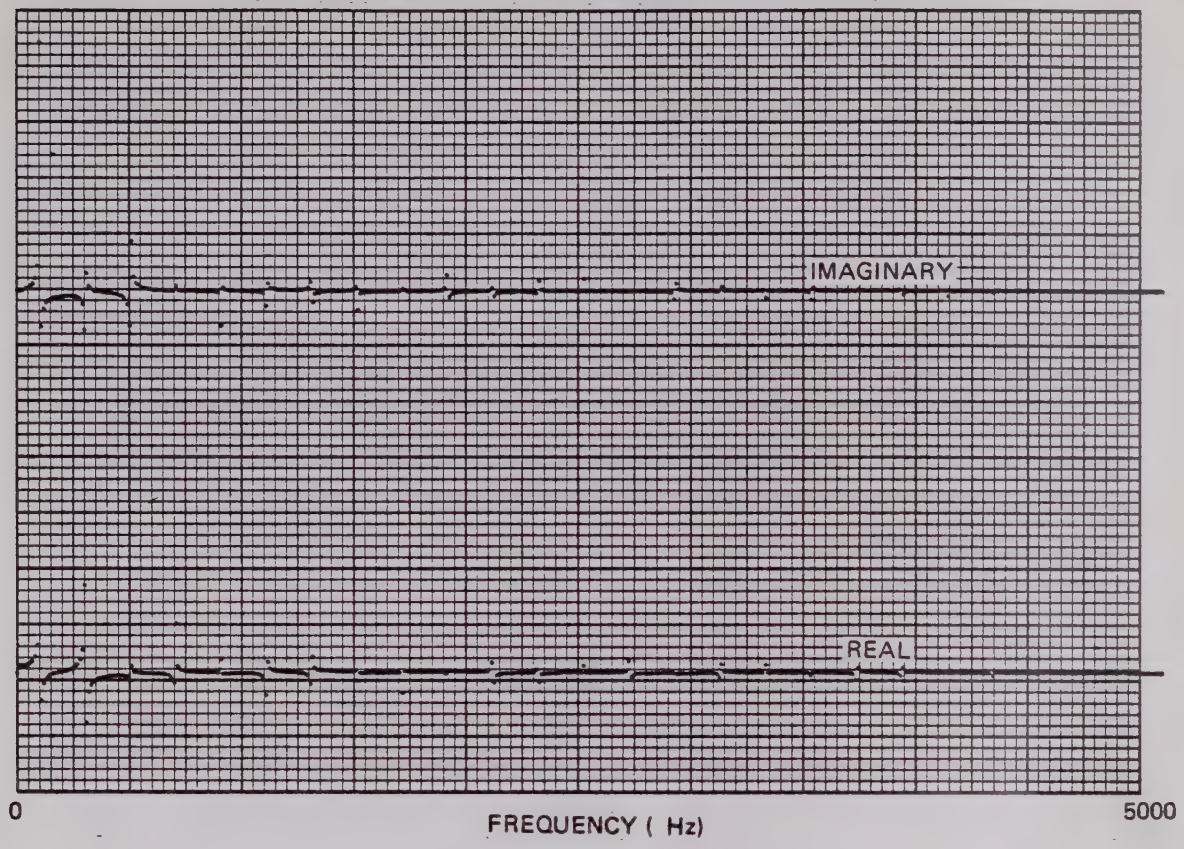


Figure 60. Function 8 – Inverse Transform Of Output Memory

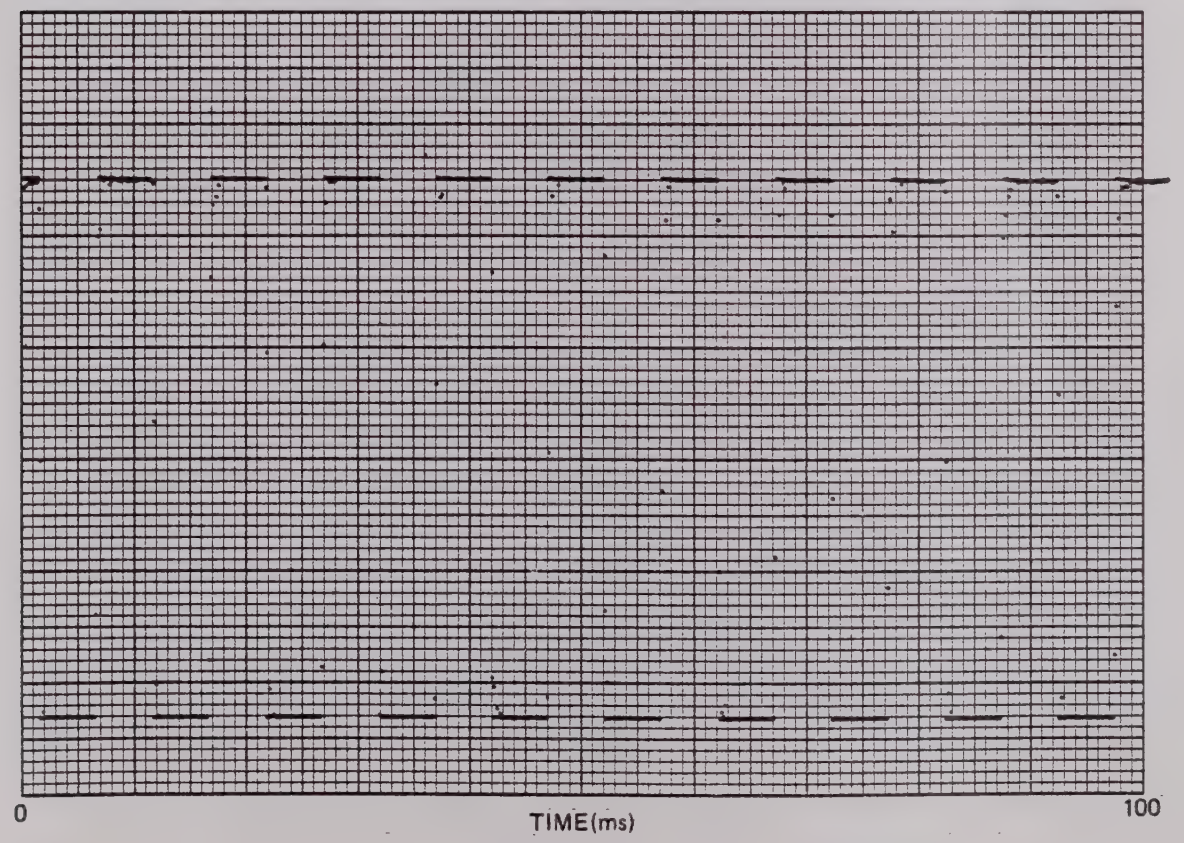


Figure 61. Input Memory — 6kHz RANGE

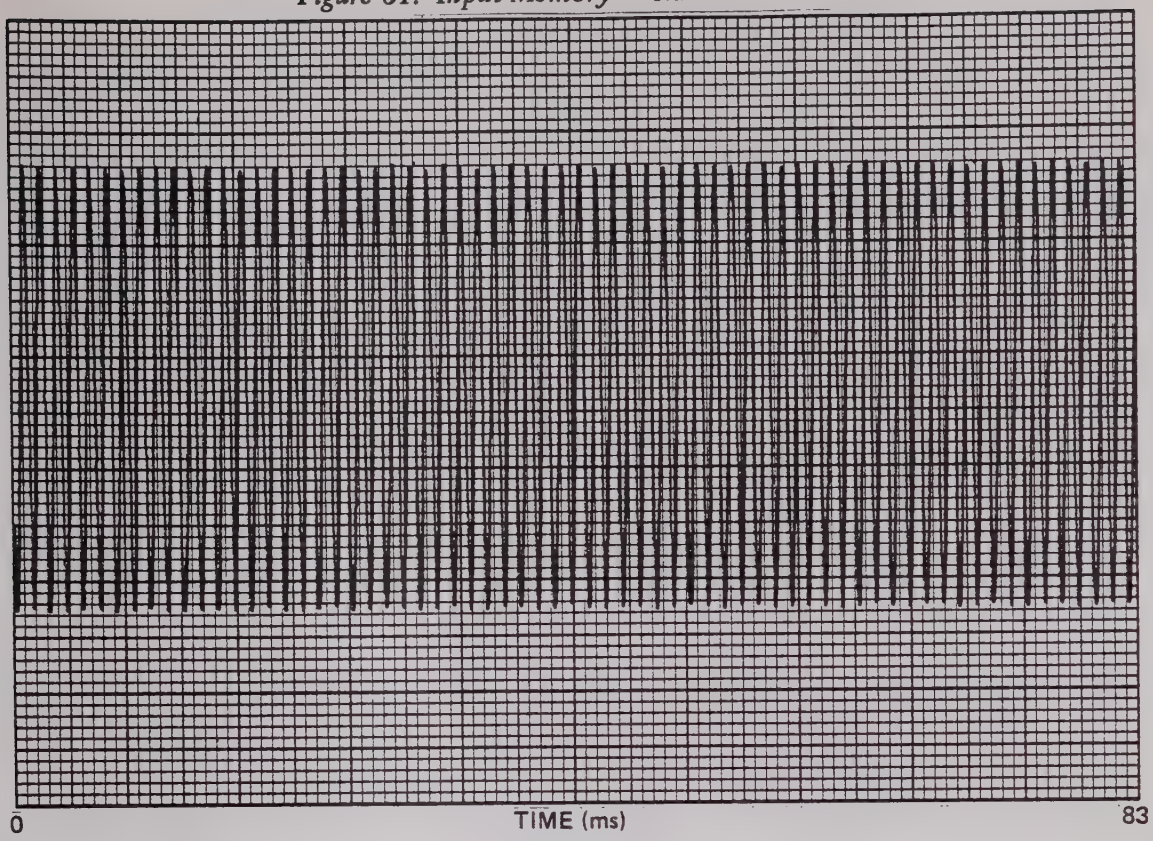
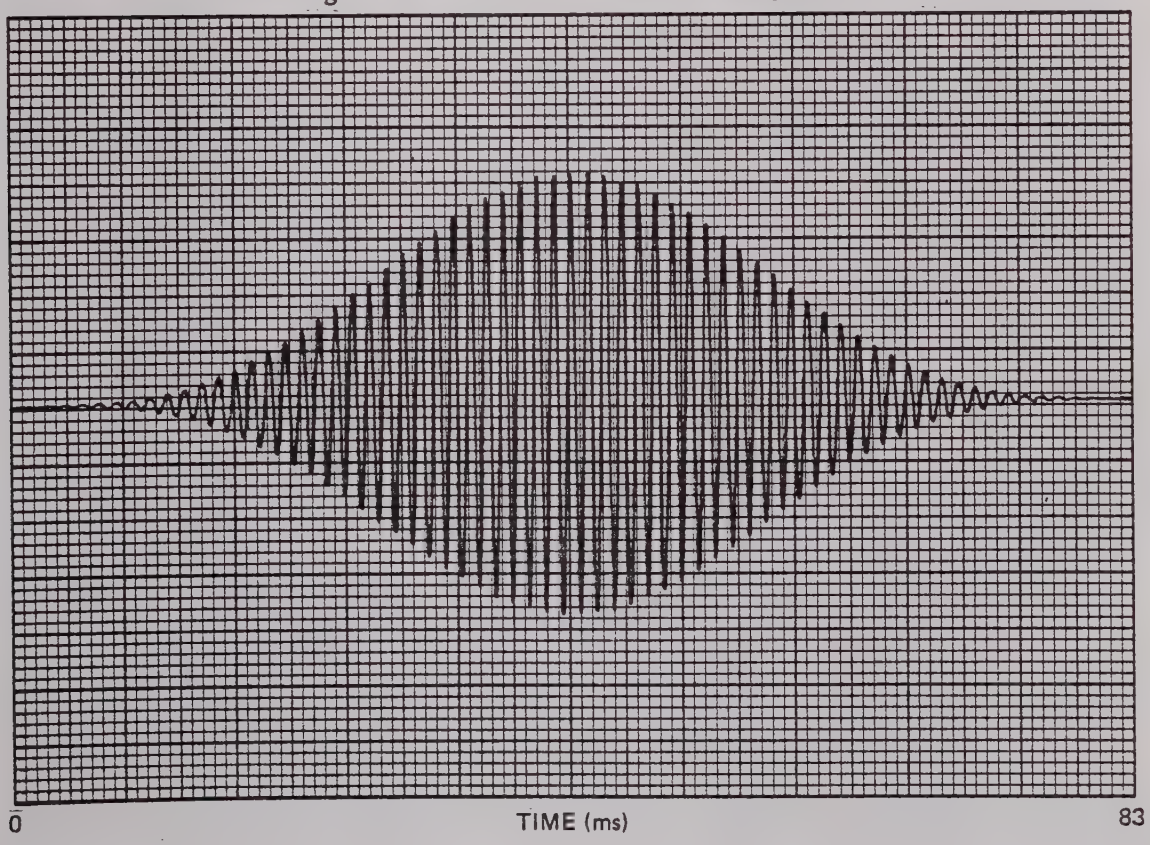


Figure 62. Function 8 — Gaussian Weighting



FUNCTION 9 – AUTO CORRELATION

The autocorrelation function, $R_{xx}(\tau)$, describes the general dependence of the values of a time series of data at one time with respect to the values at another time. An estimate for the autocorrelation between the values x(t) at times t and t + τ may be obtained by taking the product of the two values and averaging over the observation time T. The resulting averaged product will approach an exact autocorrelation function as T approaches infinity.

The quantity $R_{xx}(\tau)$ is always a real-valued even function with a maximum at $\tau = 0$, and may be either positive or negative. $R_{xx}(\tau)$ is independent of phase.

Autocorrelation and the power spectrum are a Fourier transform pair.

$$\mathfrak{F}[R_{xx}(\tau)] = G_{xx}(\omega); \quad \mathfrak{F}^{-1}[G_{xx}(\omega)] = R_{xx}(\tau)$$

Function 9 uses this relationship to compute autocorrelation. First the power spectrum is computed and then an inverse Fourier transform is performed to give the autocorrelation function.

Input data from Channel A is simultaneously loaded into memory A and memory B. The first half of each memory is loaded with data and the remaining half is loaded with zeros to minimize circular correlation effects. However, the zeros in one half of each memory produce a linearly decreasing number of products being summed to produce the resulting function. To compensate for this phenomenon a front panel selectable correction factor can be added to the output data to bring the output display to full scale. Weighting is automatically removed when performing this function as the output results would be distorted.

A Forward transform is taken usually with averaging. During buildup, the display observed is the rectangular coordinates of the forward transform with the real part in the right array and the imaginary part in the left array. Data flow for a Function 9 is given in Figure 63. Once the averaging is complete, the display automatically pops over to the correlation function.

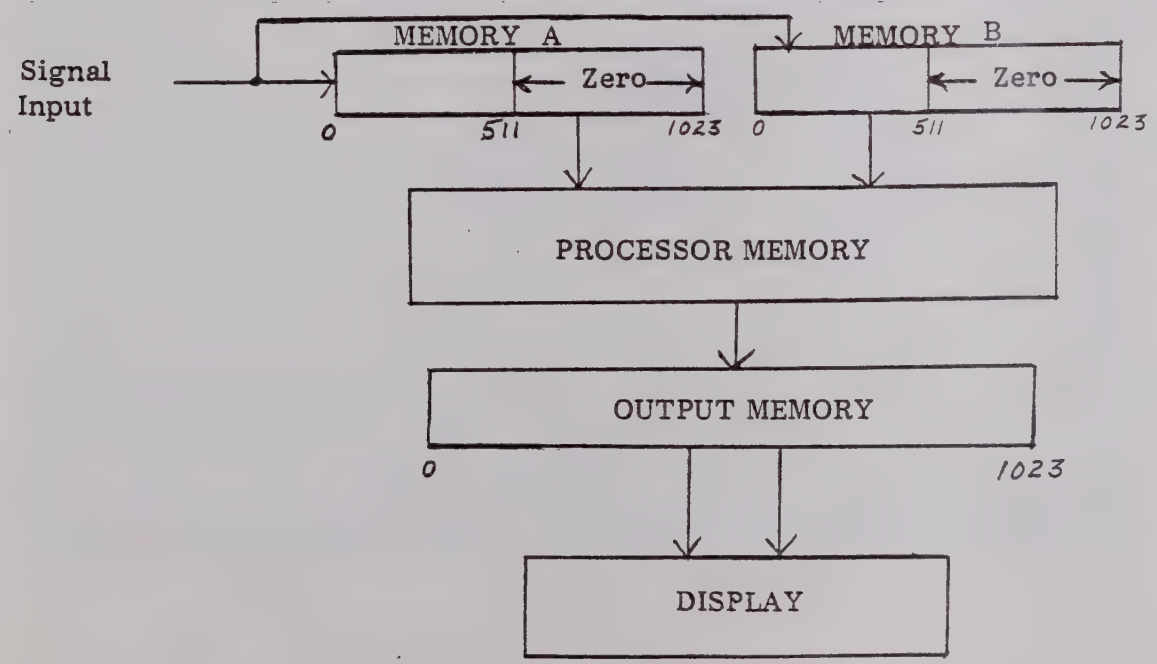


Figure 63. Data Flow For Function 9

FUNCTION 9 – AUTOCORRELATION

The Autocorrelation Function is performed by first doing a Fourier Transform on the signal, either with or without averaging, and then taking the Inverse Fourier Transform after the average has been completed. The Autocorrelation function gives a direct indication of how a signal relates to itself as a function of time.

If the Autocorrelation function is performed on the SD360 without averaging, the display alternates back and forth between a spectrum analysis real and imaginary parts and the correlation estimate (Figures 65 & 66). If averaging is performed, the operator views an averaged spectrum building up until the selected number of averages has been accumulated (Figure 65). At the completion of the average, the display automatically changes to the time domain correlogram (Figure 67). During the spectrum analysis only the real portion is displayed (righthand side of the display).

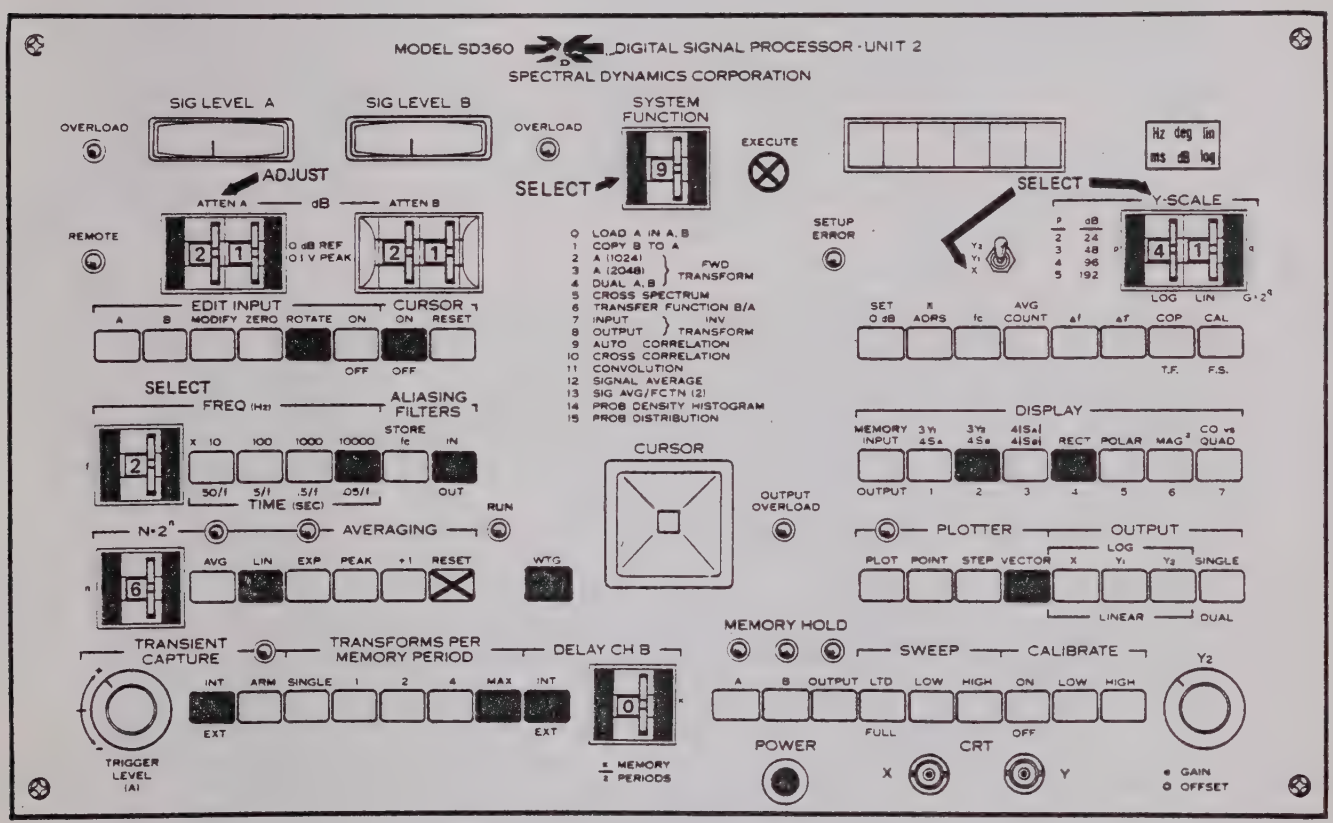


Figure 64. Function 9 Diagram

1. Using the Function 9 diagram, select the front panel controls designated ■ and set the thumbwheel switches to the number designated by [X].

2. Apply the data signal to Channel A and adjust the attenuator if an input overload condition occurs.

3. Momentarily, press the Averaging RESET button and push the EXECUTE control.

4. Adjust the q switch to introduce output gain for the display.

5. Momentarily, push the AVG control. When the averaging process has been completed, the Autocorrelation function will appear on the display.

The Autocorrelation function is aways a real valued, even function with a maximum amplitude occurring at $\tau = 0$. The 0 time position is in the center of the display and the Autocorrelation function can exhibit either a positive or a negative maximum amplitude. The cursor dot can be positioned to correspond to this 0 time location. The actual time can be read out on the LED readout with the axis toggle switch in the X position. Momentarily, press the $\Delta \tau$ control and observe the correlation delay increment that is displayed in the LED readout. This increment is established by the sampling rate that is used for the selected frequency analysis range. Momentarily, press the CURSOR RESET button. The LED readout will display the negative time limit for the selected correlation window.

If the Input Data signal is a sine wave, the Autocorrelation function will show up as a cosine wave with a linear taper from the maximum amplitude at $\tau = 0$ to 0 amplitude at the lefthand and righthand edges of the display (Figure 68). This linear taper is the result of utilizing half of Input Memories A and B for the data signal while the other half is loaded with zeroes to eliminate the possibility of introducing circular correlation errors. Since this taper is a known predictable phenomena, it can be corrected by selecting display $3Y_1 4S_A$ and re-averaging the input data signal (Figure 69). This adds a correction factor to the output data to bring the display to full scale over the total correlation window. The 22 cells at the beginning and end of the window are not corrected when the correction is introduced, since by correcting very small amplitudes there would be little confidence in the data. The last 22 cells on either end will actually contain the linear real and imaginary parts at the computed power spectrum (F9) or cross-spectrum (F10).

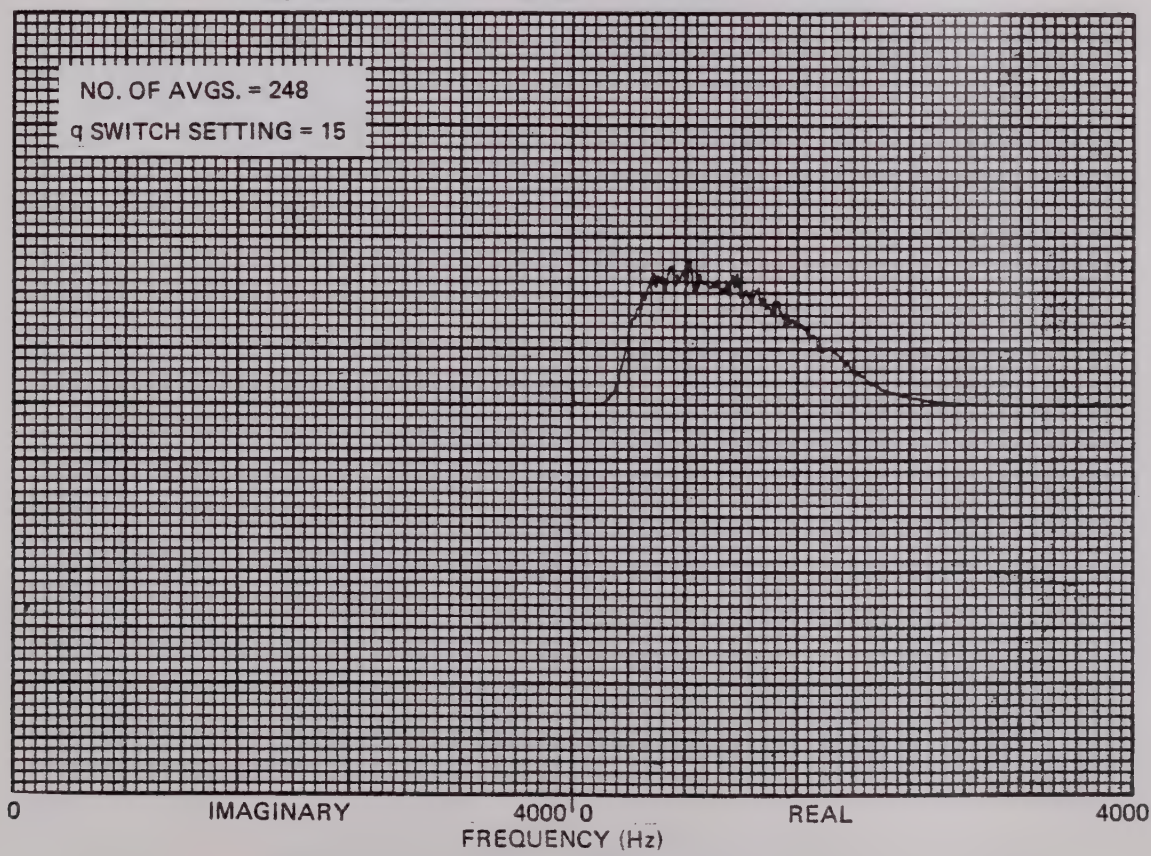


Figure 65. Function 9 – Averaged Spectrum

Figure 66. Function 9 — Time Domain Correlogram

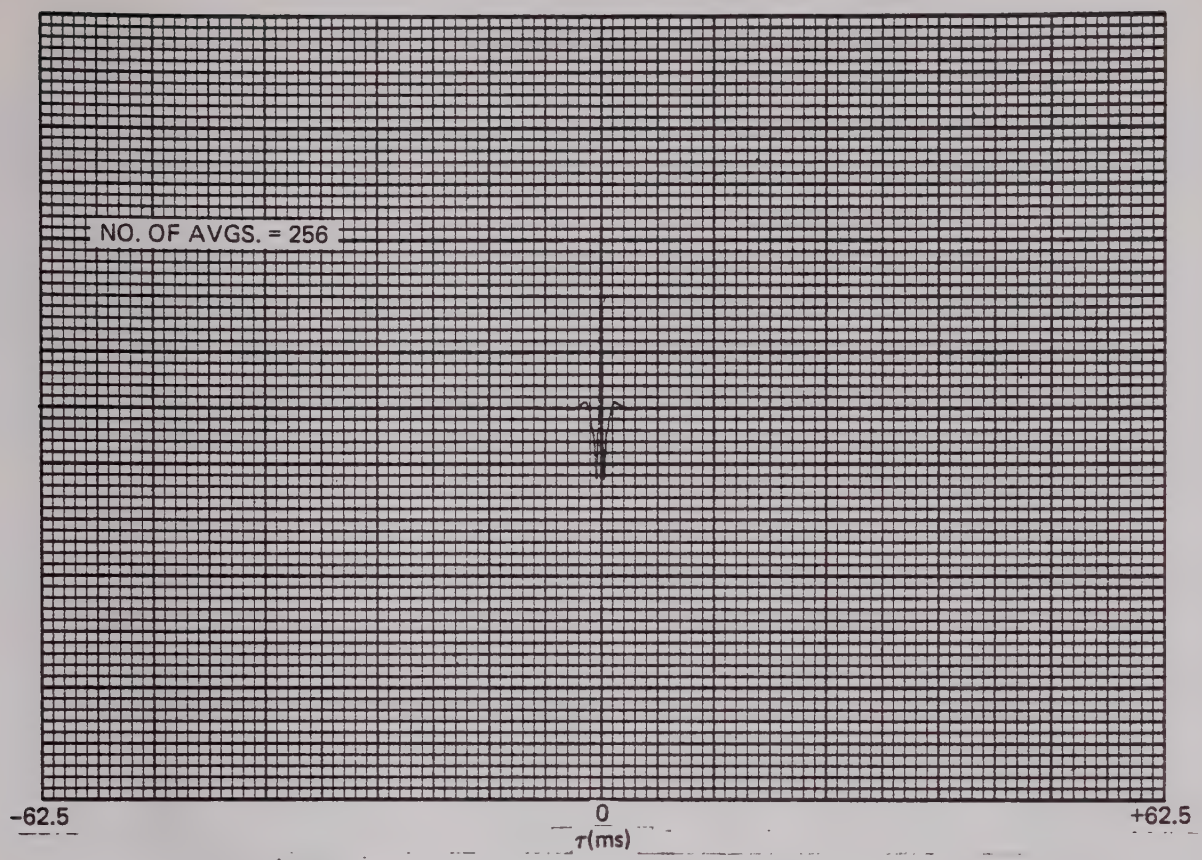


Figure 67. Function 9 — Time Domain Correlogram

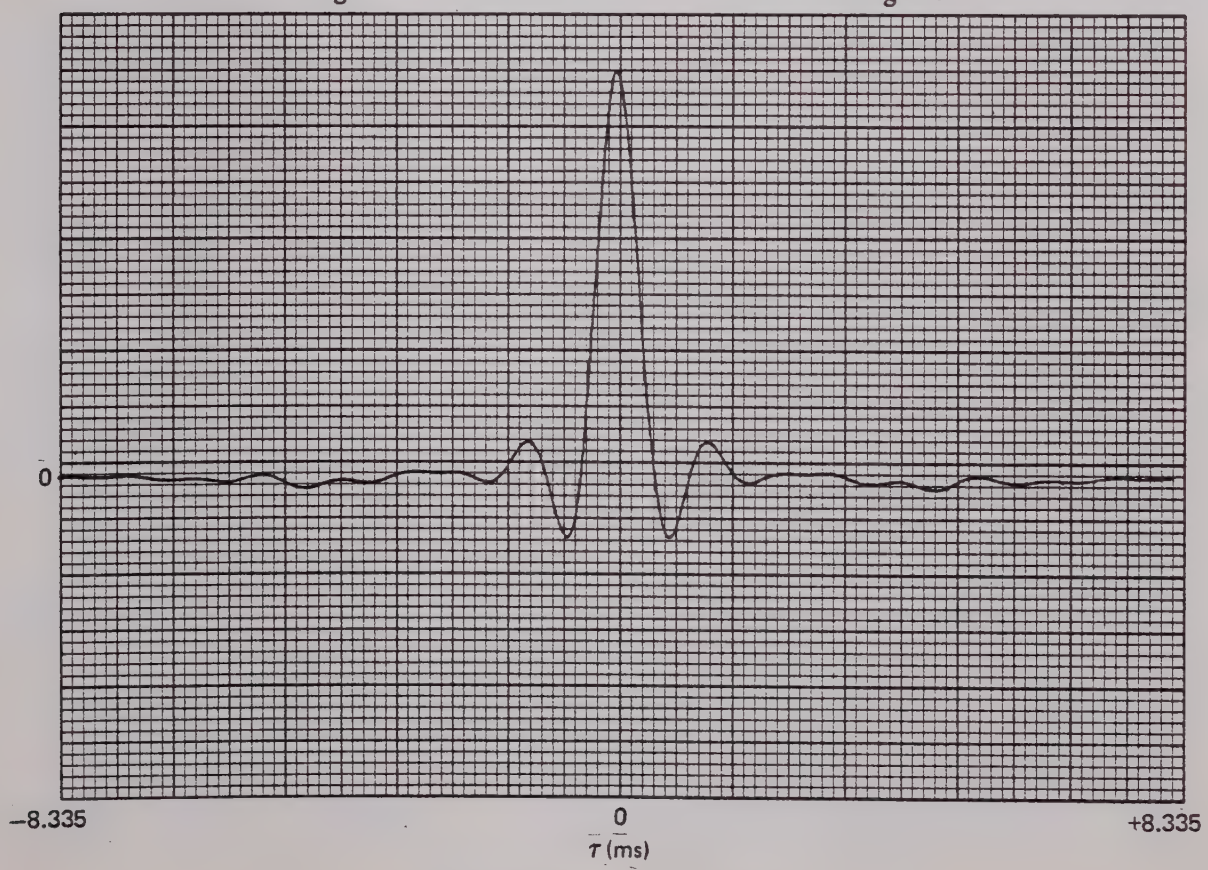


Figure 68. Function 9 — Auto-Correlation (Uncorrected)

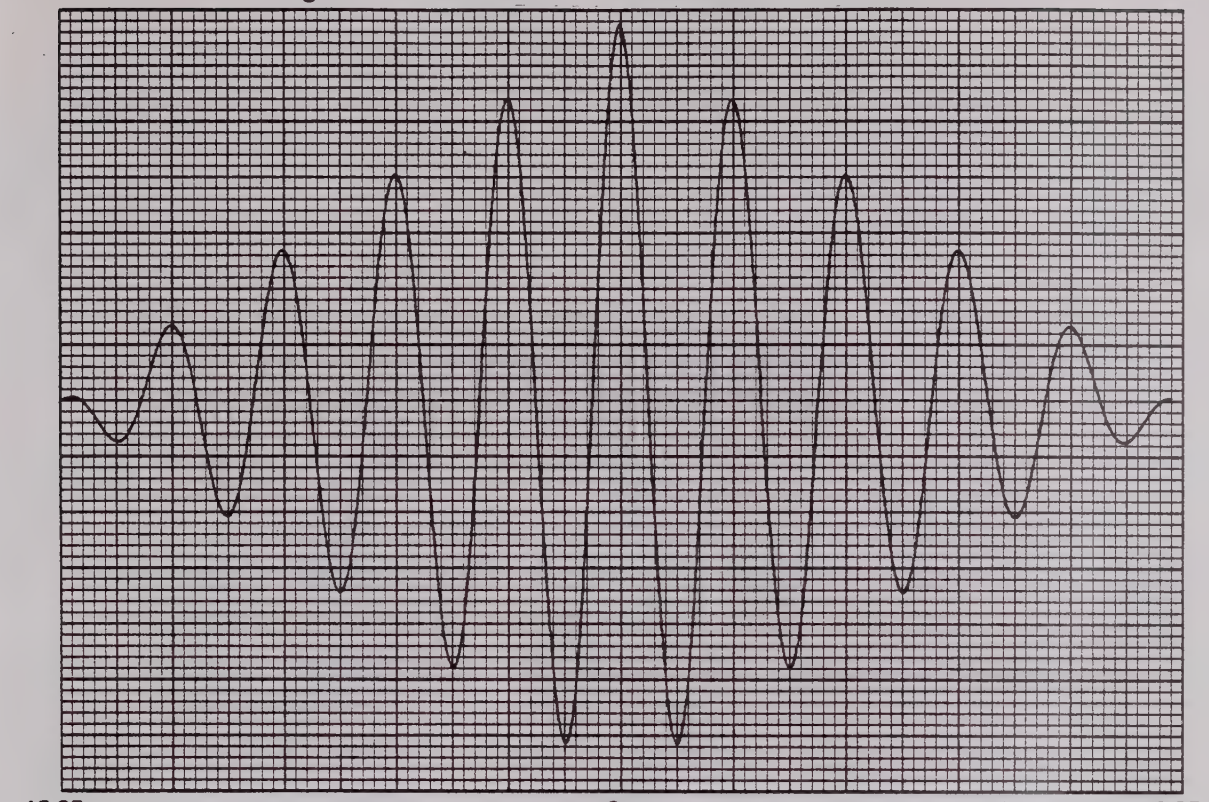
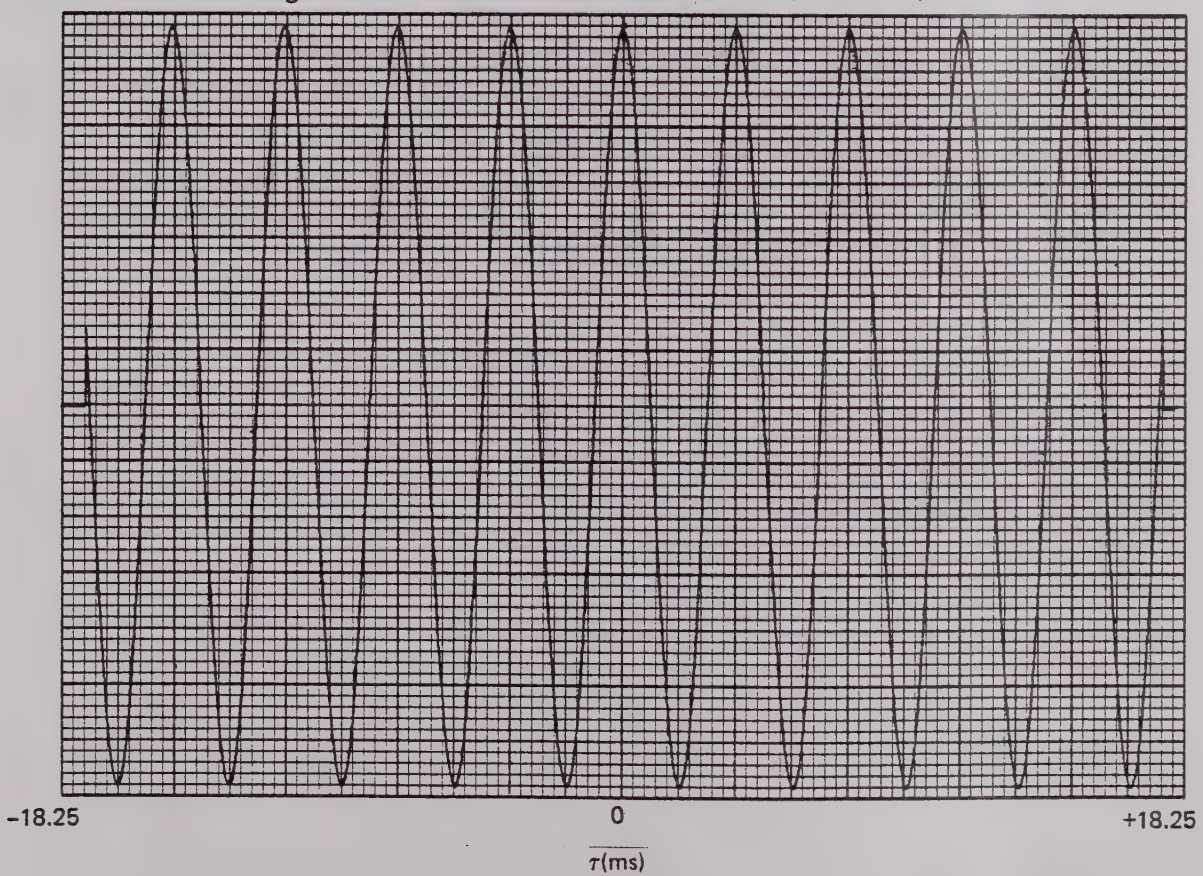


Figure 69. Function 9 — Auto-Correlation (Corrected)



FUNCTION 10 – CROSS-CORRELATION

Cross-correlation, $R_{yx}(\tau)$, shows the similarities between two signals as a function of time shift τ. The cross-correlation and the cross-spectrum functions are a Fourier transform pair.

$$R_{yx}(\tau) = \mathfrak{F}^{-1}[G_{yx}(j\omega)]; \; G_{yx}(j\omega) = \mathfrak{F}[R_{yx}(\tau)]$$

The cross-correlation function is a real-valued function which may be either positive or negative. It is phase sensitive and is not in general an even function of τ. Cross-correlation for two different signals may be estimated in the same general manner as autocorrelation for the same signal.

Cross-correlation is used in the determination of time delays between two signals which may be buried in noise or which may consist of broadband random noise. Cross-correlation may be used to determine a system impulse response. This would be done by exciting a system with flat broadband random noise, performing a cross-correlation between the input and the output and storing in the output memory this computed cross-correlation function, which by definition is the system's impulse response.

Function 10 is obtained by computing the inverse transform of the cross-spectrum of data from channels A and B. The input memories are loaded half with data and half with zeros, as in Function 9, to avoid circular correlation effects. The same correction for full-scale display of output is available for this function at operator option. Again, weighting is automatically removed. If averaging is performed, only the LIN averaging mode may be used. When averaging is complete, an inverse transform of the output cross-spectrum data is performed and the results placed in output memory. Data flow for a Function 10 is given in Figure 70.

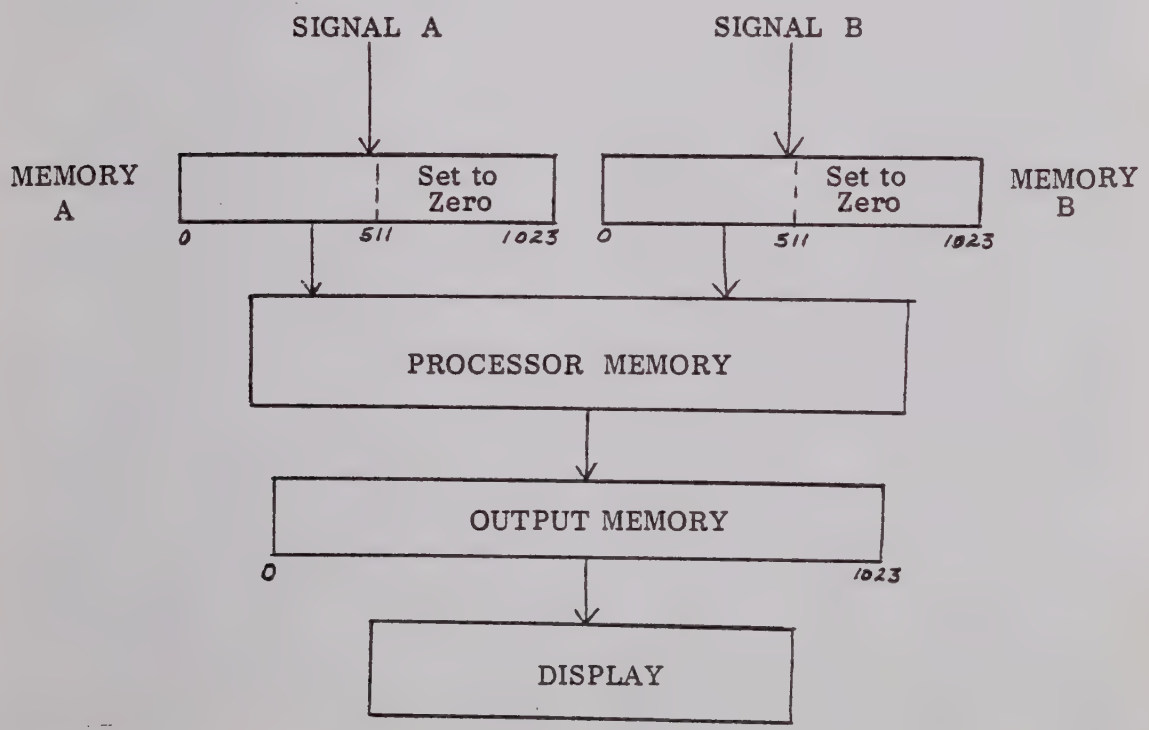


Figure 70. Data Flow For Function 10

FUNCTION 10 – CROSS-CORRELATION

The Cross-Correlation function on the SD360 is computed much the same as the Autocorrelation function; that is, by using the Fourier Transform pair properties between the cross-correlation function and the cross-spectrum function. The cross-correlation function is computed by first doing a Fourier Transform on the signal with, or without averaging, and then performing an inverse transform after the averaging is completed.

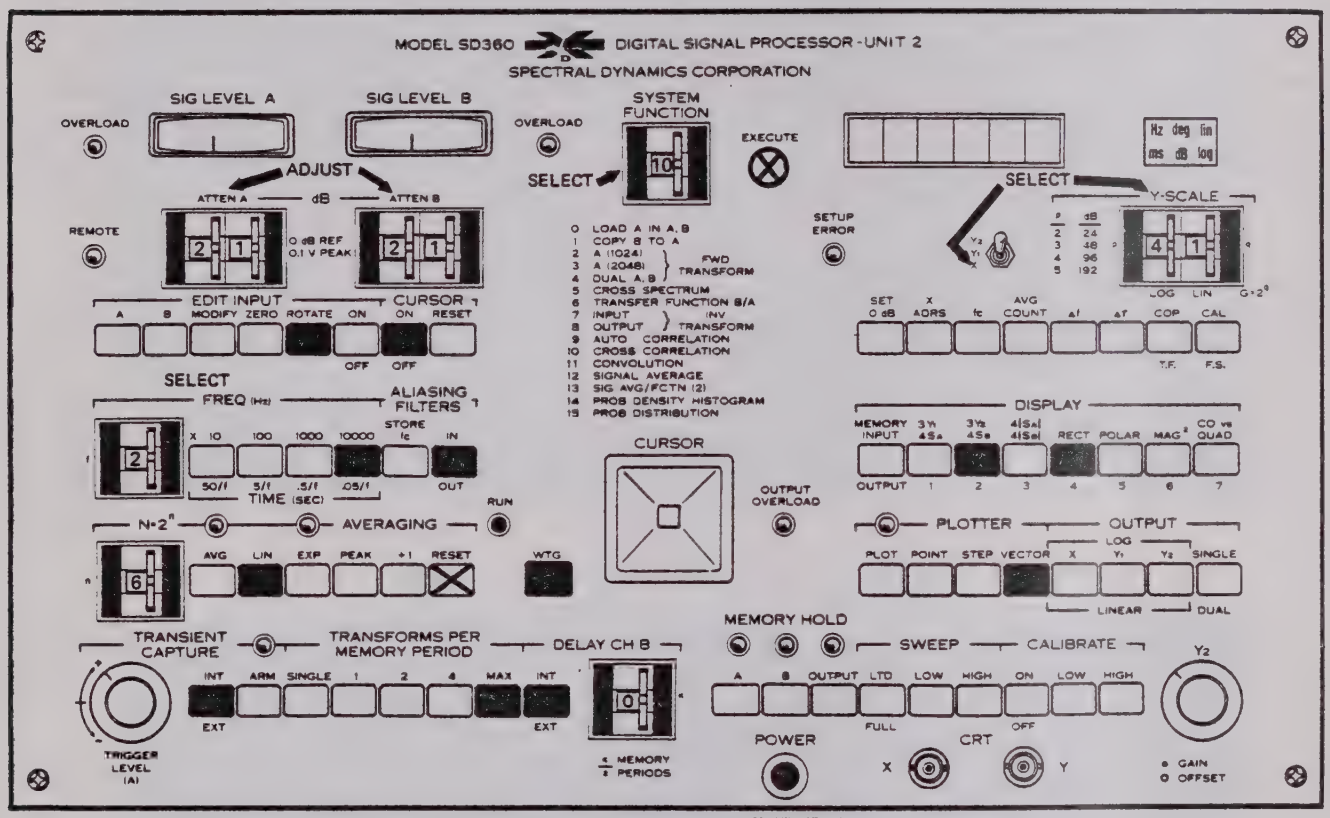


Figure 71. Function 10 Diagram

1. Using the Function 10 diagram, select the front panel controls designated by ■, set the thumbwheel switches to the number x designated by [x].
2. Apply the data signals to be correlated to CH A and CH B input and adjust the attenuators if an overload condition exists.
3. Momentarily press the AVERAGING RESET push button.

4. Push EXECUTE.

5. Adjust the q switch to introduce output gain for the display without overload.

6. Momentarily press the AVG control. When the averaging process is completed, the cross-correlation function will appear on the display (Figure 72).

7. To change the display parameters, the procedure is the same as for the autocorrelation function.

The display for the cross-correlation function is presented in much the same manner as the autocorrelation function. If the cross-correlation function is performed without averaging, the display will alternate between a spectrum analysis of real and imaginary parts and the correlation estimate. During an averaging cycle the operator can view the real part of a cross-spectrum on the righthand side and an imaginary part on the lefthand side (Figure 73). The operator views the averaged spectrum building up until the selected number of averages has been accumulated. At the completion of the average the display switches to the time domain correlogram (Figure 74).

The same display parameters that were discussed in the autocorrelation function also apply to the cross-correlation function. The 0 time position is in the center of the display. The operator has the choice of displaying the corrected or the uncorrected correlogram. The correction is selected in the same manner as for the autocorrelation function.

If the cross-correlation is used to obtain the impulse response information from a system, it is recommended that the uncorrected cross-correlation display be used. Typically the signals are low level and a sufficient amount of output gain dialed in by the q switch must be added to expand the correlation display.

Figure 72. Function 10 — Cross-Correlation

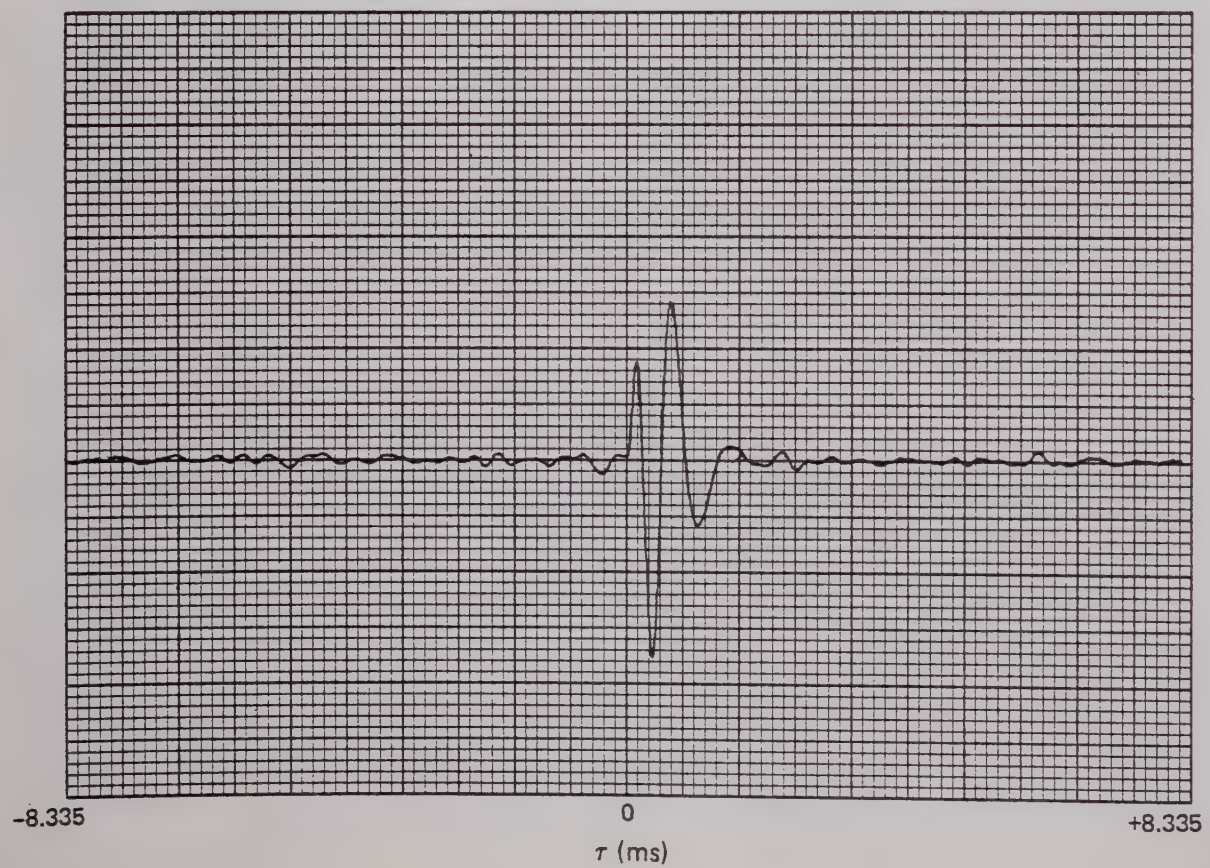
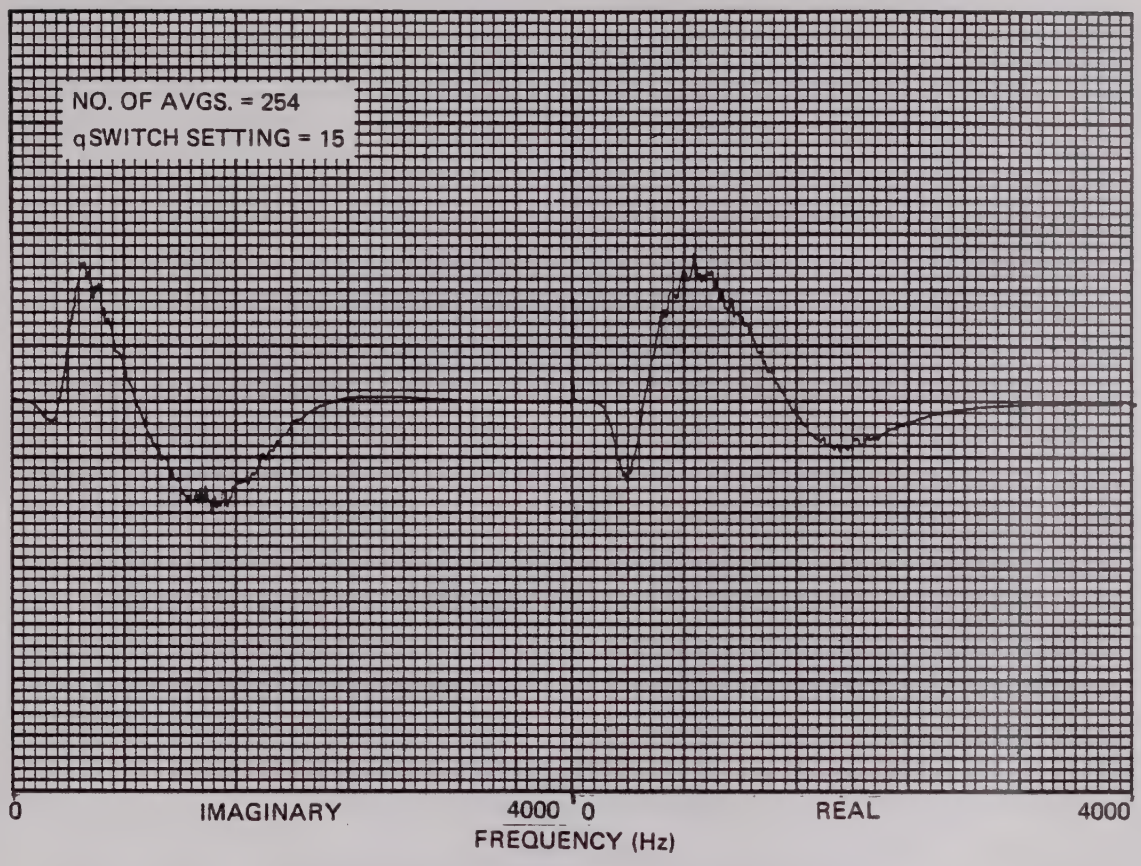
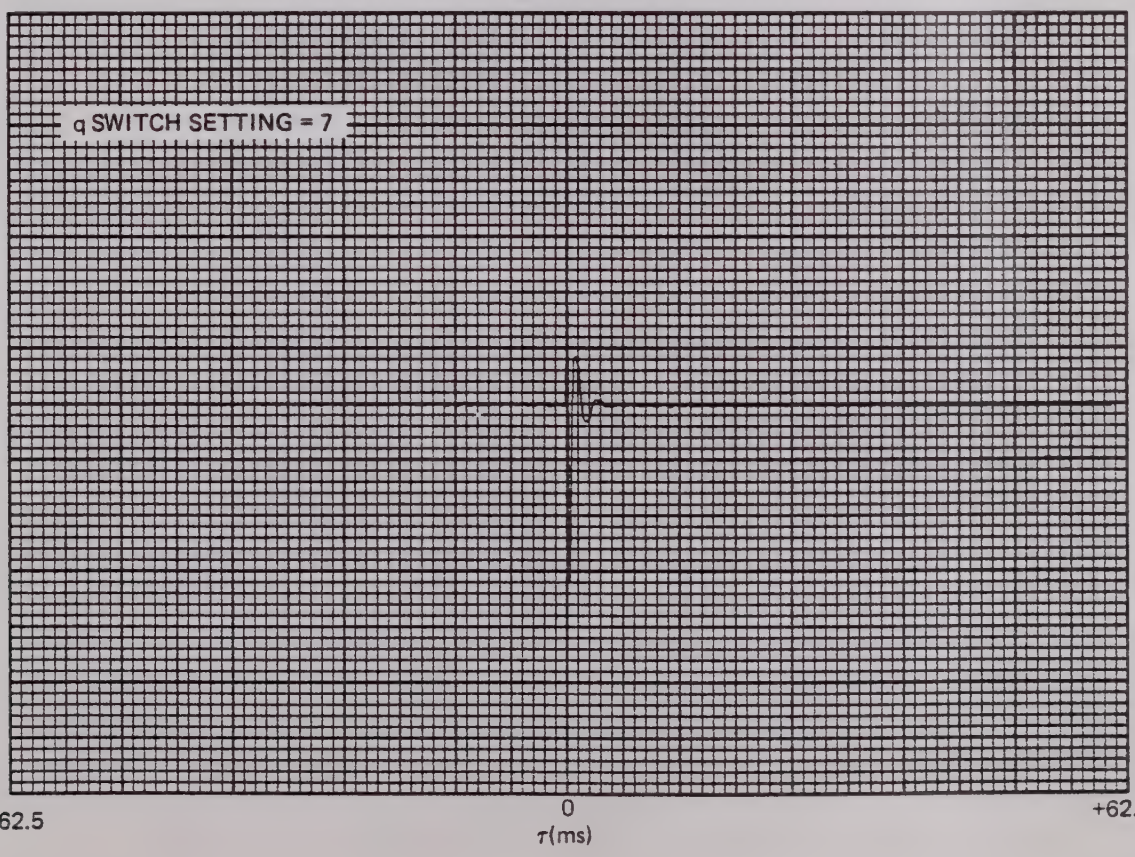


Figure 73. Cross-Spectrum During Averaging





ACOUSTIC ARRAY ANALYSIS
PRIMARILY BROADBAND
SOUND SOURCES

STEP 1 Perform a Function 4 – (G_{BB}, G_{CC}) Power Spectral Density for Sources B and C (figure 75).

STEP 2 Perform a Function 6 – ($\gamma^2 G_{BB}$, $\phi_{B\text{-}C}$) Coherent Output Power for Sources B and C (figure 76).

STEP 3 Perform a Function 6 – ($\gamma^2 G_{BB}$, $\phi_{B\text{-}C}$) Coherent Output Power; Expanded Range (figure 77).

STEP 4 Perform a Function 6 – ($\gamma^2 G_{AA}$, $\phi_{B\text{-}A}$) Coherent Output Power for Sources A and B (figure 78).

STEP 5 Perform a Function 10 – (R_{BA}) for Source A and B (figure 79).

Figure 75. Function 4 — 64 Averages

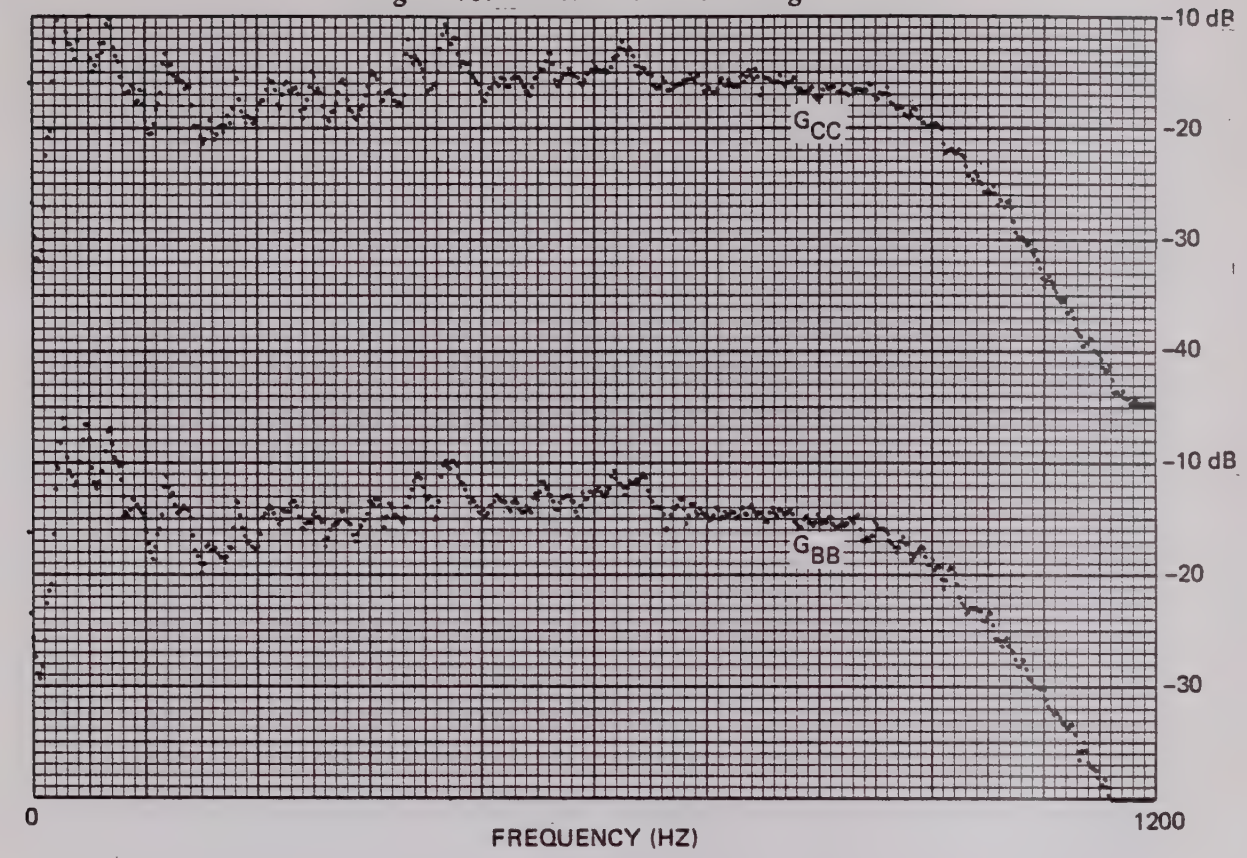


Figure 76. Function 6 — Coherent Output Power

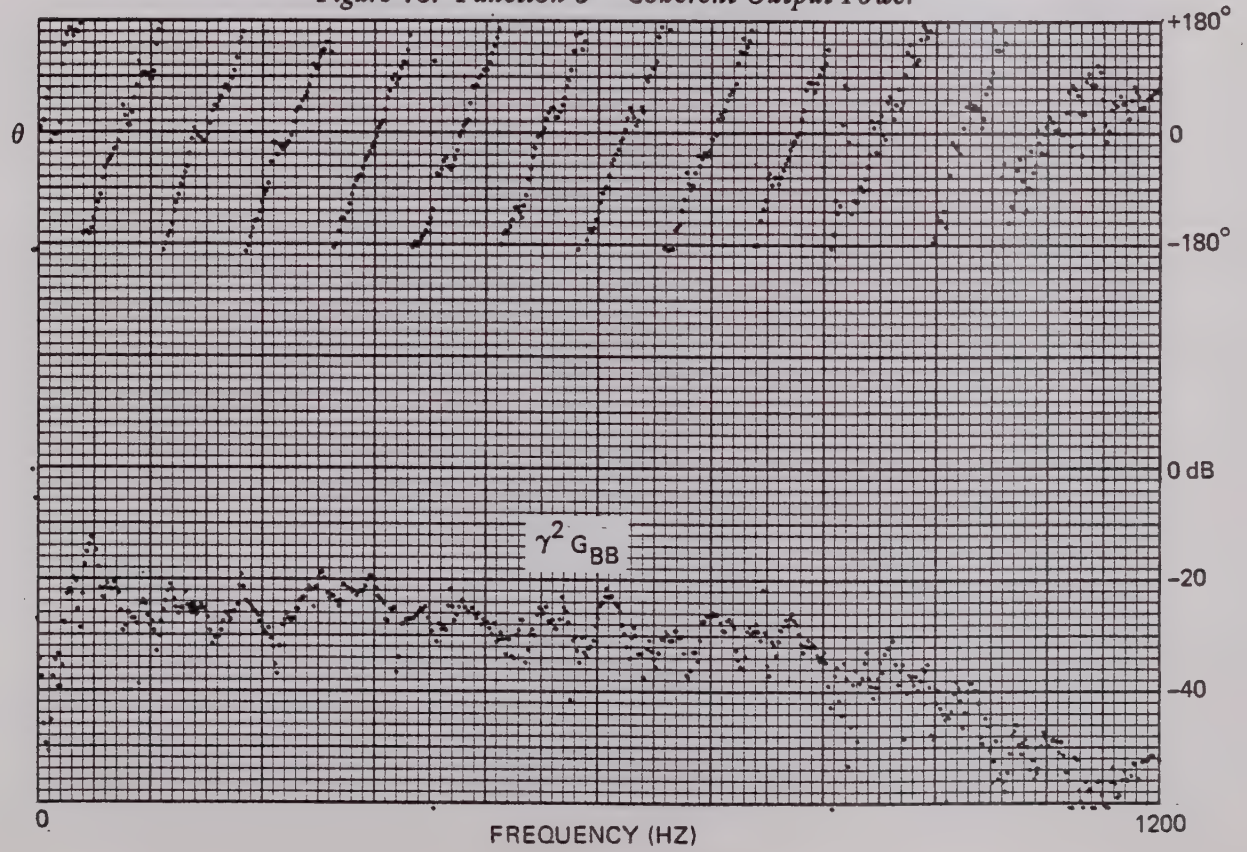


Figure 77. Function 6 — Coherent Output Power

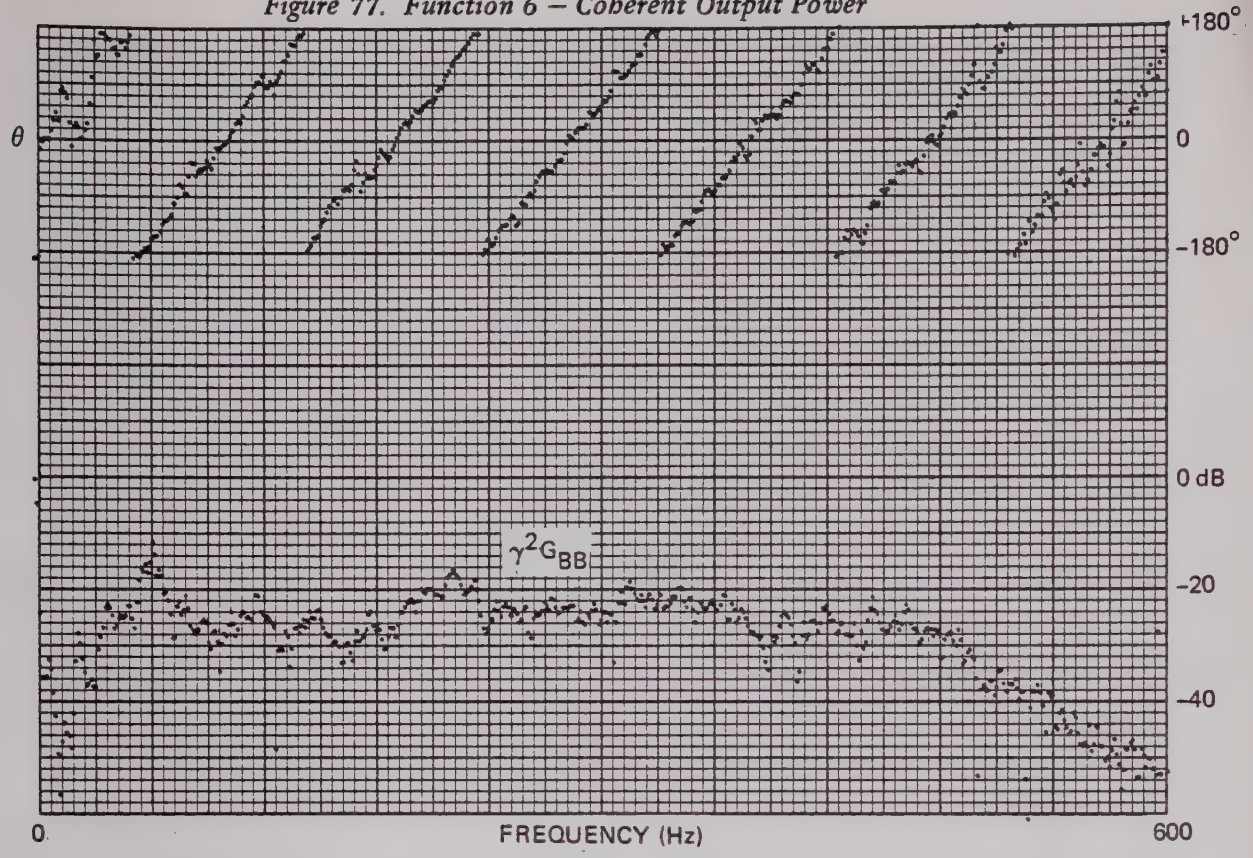


Figure 78. Function 6 — Coherent Output Power

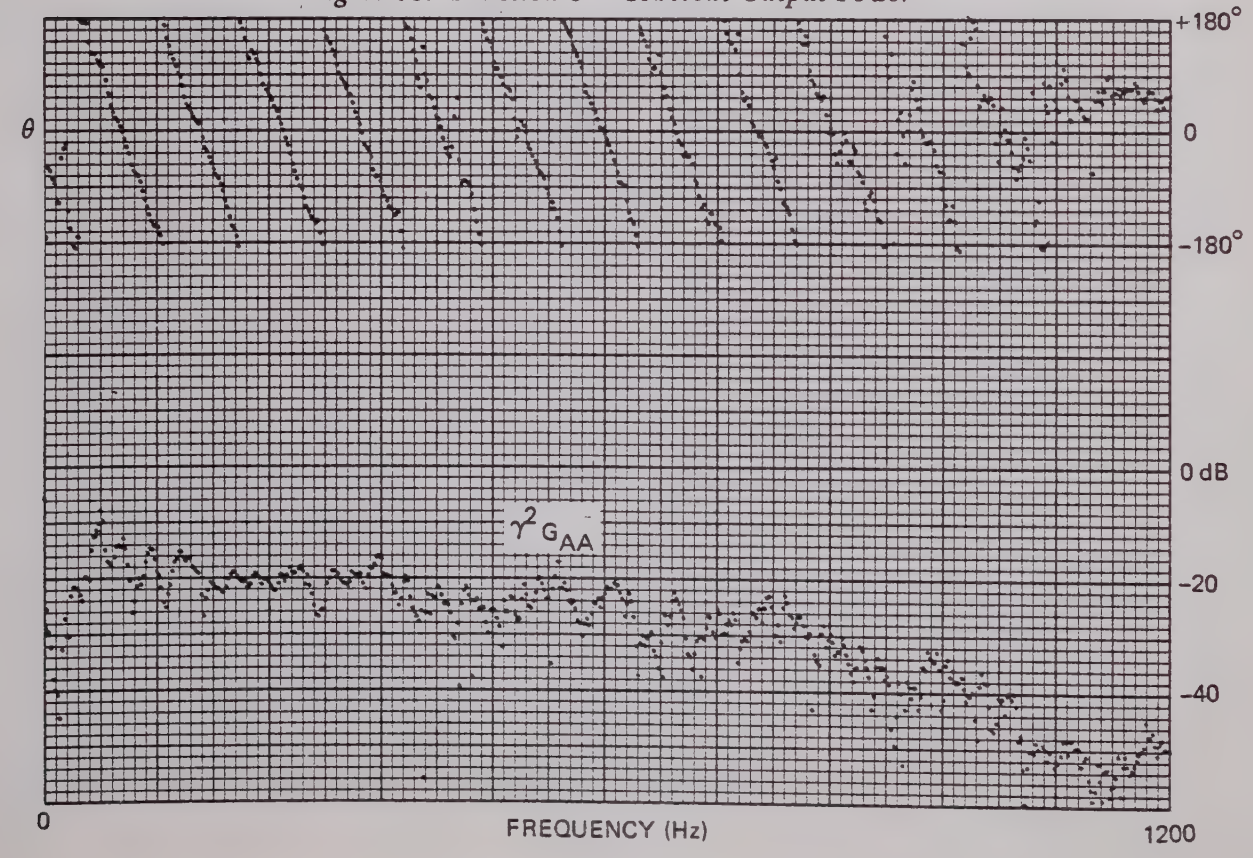
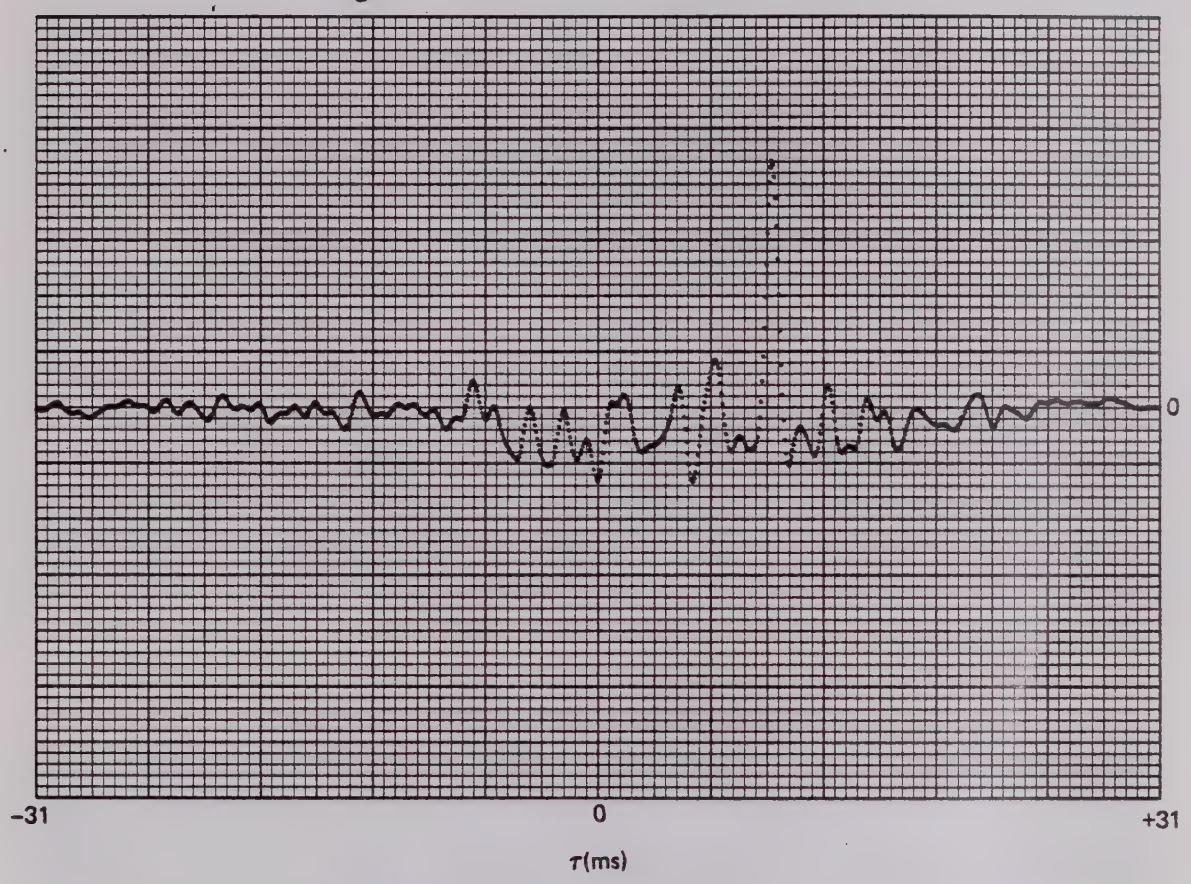


Figure 79. Function 10 — Cross-Correlation



The sequence of recordings (Figures 76—79) show the characteristics of two hydrophone signals. The phase display of Figures 76—78 showed a continuous phase shift with frequency. This is characteristic of signals which are primarily broadband noise and have a time delay between them. Note how clearly the cross-correlation function in Figure 79 shows a time delay between signals of 9.4 milliseconds.

FUNCTION 11 – CONVOLUTION

Convolution, $C_{yx}(t)$, is used in determining a specific time response. Linear systems theory states that if the impulse response of a system is known, a time history representing the input to the system can be convolved with the system's impulse response to give the actual time waveform output of the system (Figures 80-82). Convolution is defined as follows:

$$y(t) = \int_{-\infty}^{\infty} x(t)h(t-\tau)d\tau \qquad x(t)*h(t) = \int_{-\infty}^{\infty} x(\tau)h(t-\tau)d\tau$$

Convolution and the convolution spectrum are a Fourier transform pair. Convolution is the inverse transform of the convolution spectrum.

$$C_{yx}(t) = \mathfrak{F}^{-1}\,[C_{yx}(j\omega)]$$

To compute convolution, Function 10, Cross-Correlation may be used to compute the impulse response of the system. The basic difference between the convolution function and the cross-correlation function for identical data signals should be kept in mind when performing Function 11. The basic difference is that the cross-correlation function will always have to be symmetrical about tau = 0 with a maximum value right at the center. If the data signals are at all correlatable, naturally identical signals would be, their correlation function will peak at the center. Since convolution is actually describing a time waveform, its position will not be fixed on the time axis and data will move about like a normal waveform. If only one signal in input memory were to be rotated and a cross-correlation performed, the correlogram although not generally still symmetrical would peak up in the middle at regular intervals. The convolution function would exhibit much less symmetry.

*Figure 80. System Output = h(t) * S(t)*

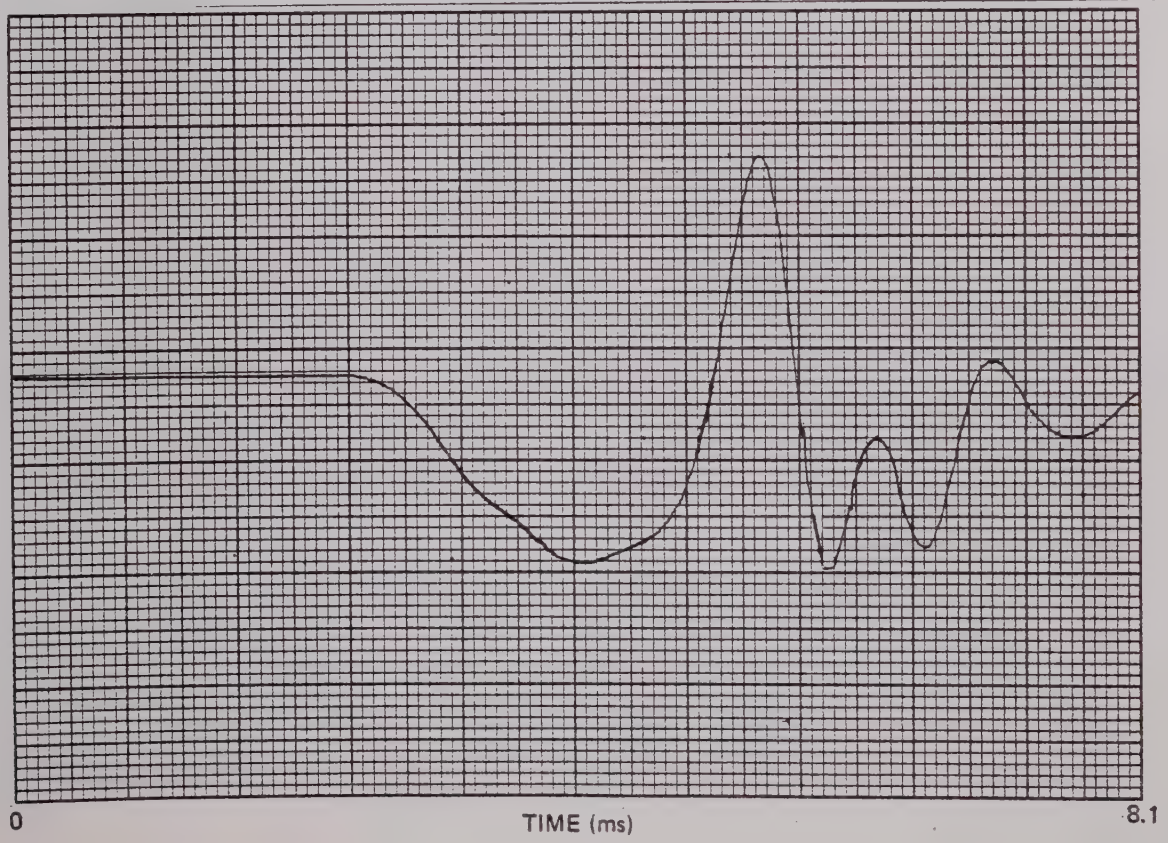


Figure 81. System Input = S(t)

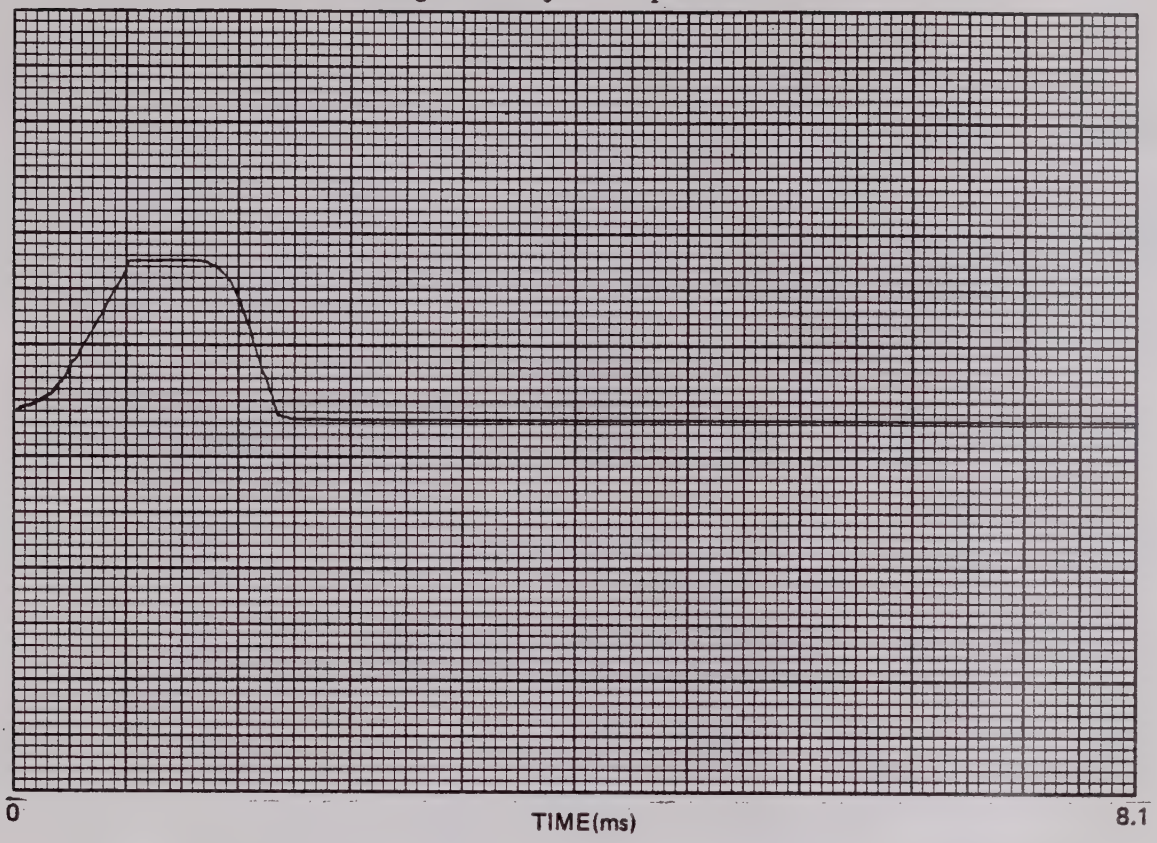
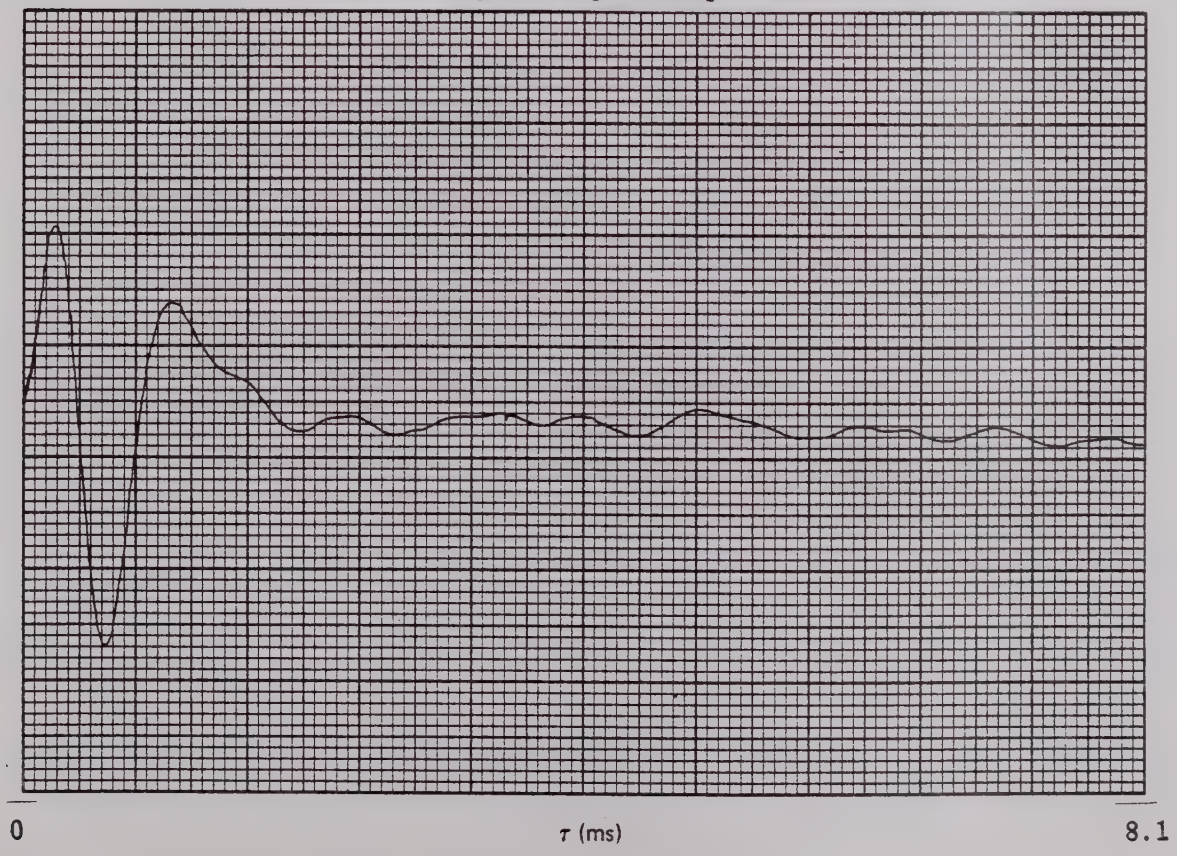


Figure 82. System Impulse Response = h(t)



FUNCTION 12 – SIGNAL AVERAGE

Signal averaging is used to recover periodic signals from a total noisy signal. For example, if a pulse train is buried in noise, the entire pulse train can be recovered through the use of signal averaging.

In Function 12 the time domain data from input memory A is averaged in the output memory using an external sync pulse and either internal or external sampling. The random components will tend to zero and the periodic components will add as the number of ensembles is increased. Data may be continuously displayed while averaging is in progress. Figure 84 is a display of the input data and the averaged waveform after performing a Function 12.

Signal Averaging

(Requires external sampling control or external sync to synchronize data with memory)

Linear Averaging

$S_{ij} = S_{i(j-1)} + x_{ij}$ stored in output memory

$A_{ij} = \frac{1}{M} S_{ij}$ displayed by the output section

$j = 1,2,\ldots,N$ ensemble index

Exponential Averaging

$$S_{i\ell} = \frac{N-1}{N} S_{i(\ell-1)} + x_{i\ell}$$

$$A_{i\ell} = \frac{1}{M} S_{i\ell}$$

$= N+1, N+2, \ldots L$ ensemble index

For Linear and Exponential Averaging

$N = 2^n$ where n = 0, 1, 2, ..., 12 on the n-switch setting

$i = 0, 1, 2, \ldots., 1023$ time index

$M = 2^m$ m such that $2^{m-1} < j \leq 2^m$

FUNCTION 13 – SIGNAL AVERAGE/FORWARD TRANSFORM

This function is used to process the spectrum of a signal whose total signal consists of data synchronous with a known reference and other non-sychronous signal contributions. An example would be a noisy environment surrounding a rotating machine of interest. By signal averaging and performing a Fourier transform of the averaged data, virtually all noise present will tend to zero without further averaging of the Fourier transform data. Figure 86 and Figure 87 show the improvement in the output by signal averaging before performing the Function 2.

When function 13 is selected, a function 12 is executed. An external sync pulse is required and the LIN Averaging mode should be selected. The averaged data is placed in output memory. A Fourier transform of this stored data is performed and selected conversion made and the results placed in output memory.

Figure 83. Function 12 Diagram

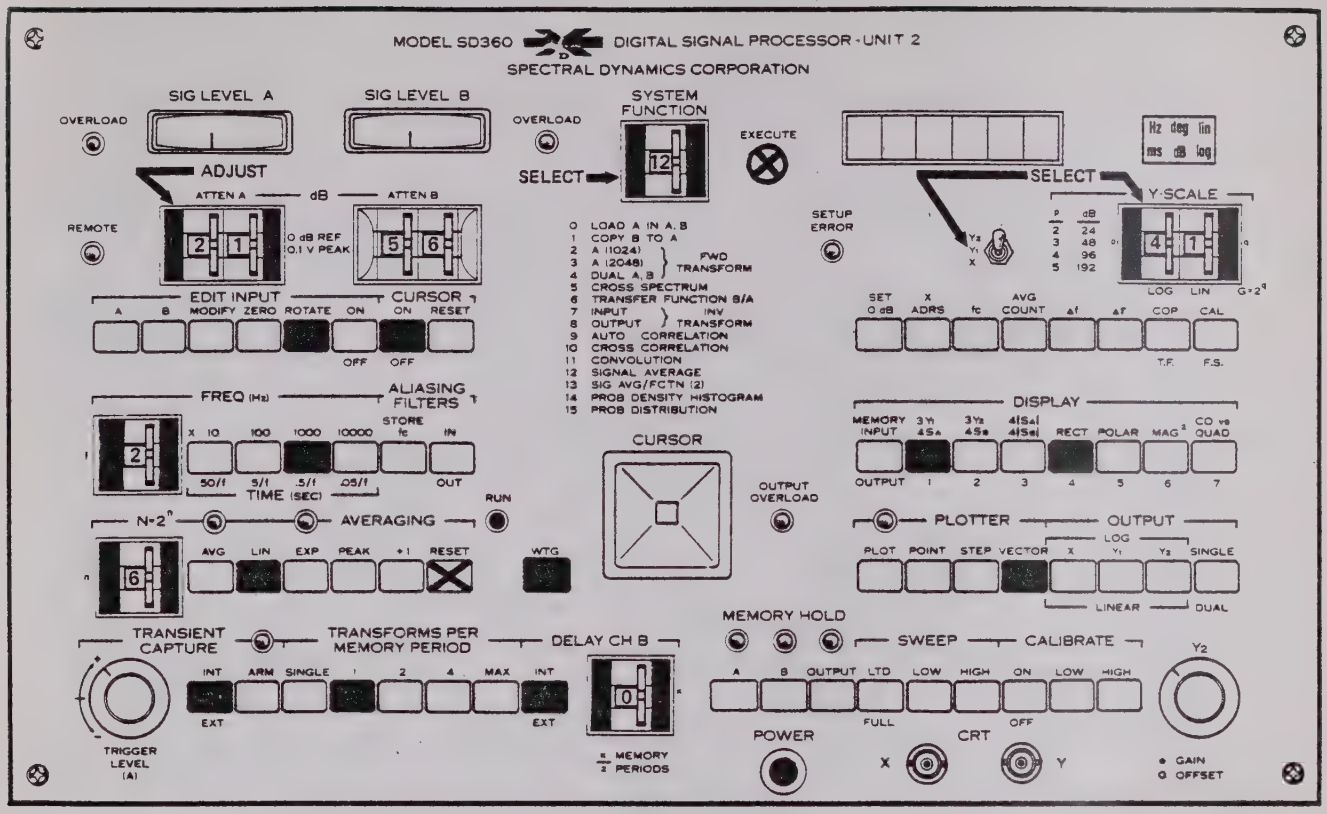


Figure 84. Function 12 — Input Data Signal And Time Averaged Waveform

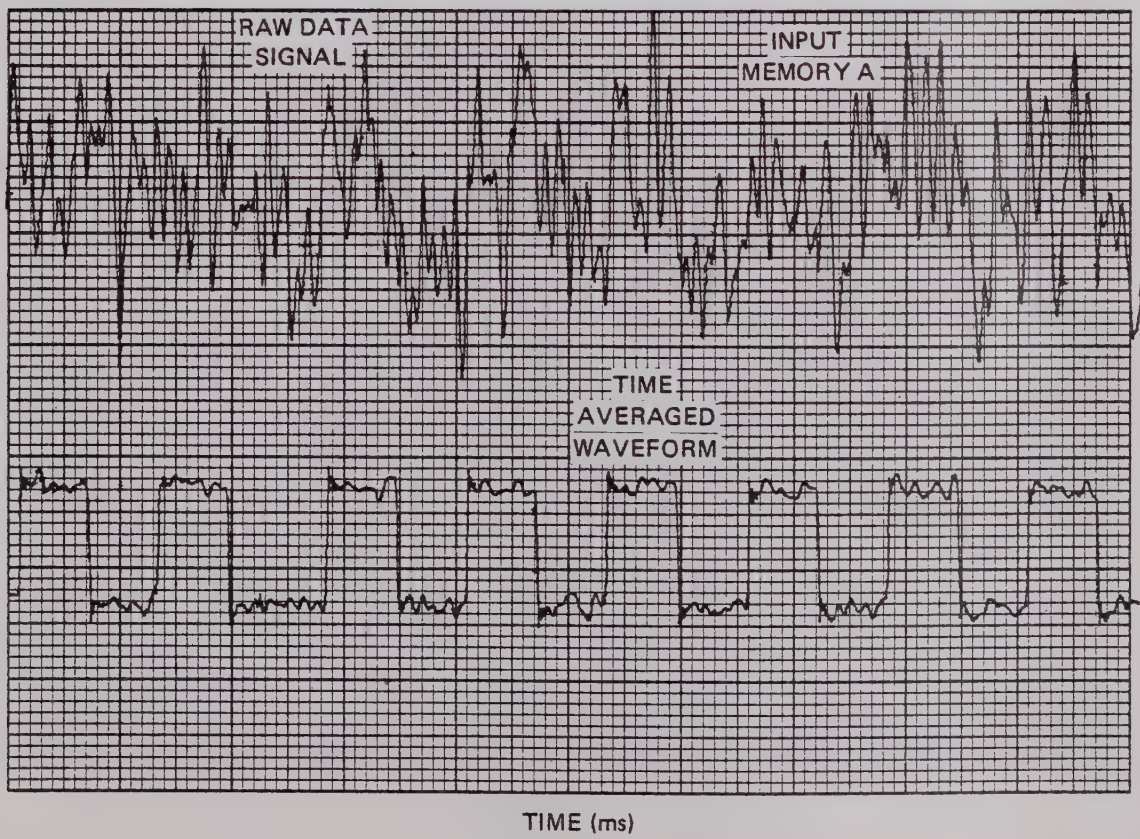


Figure 85. Function 13 Diagram

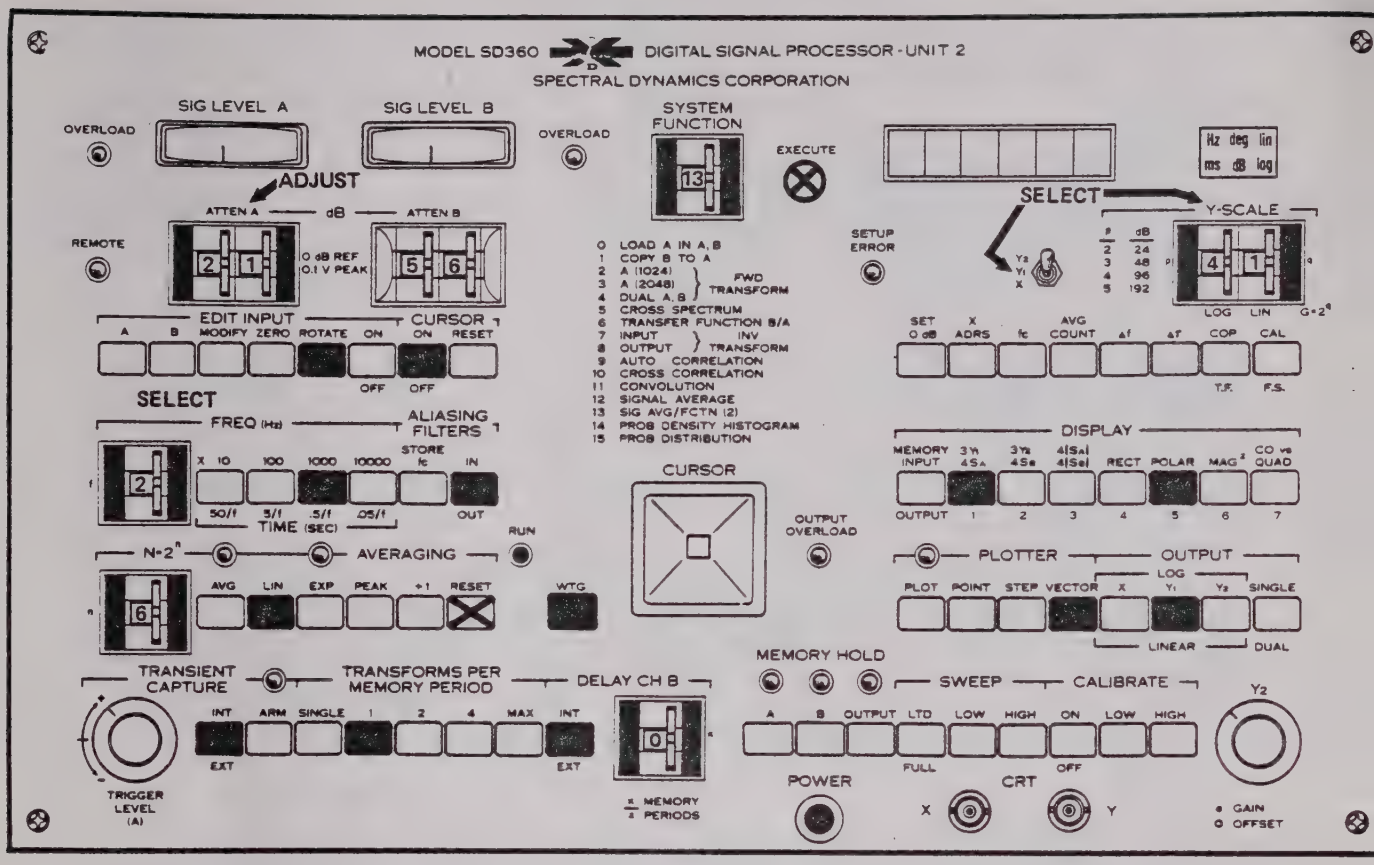


Figure 86. Function 2 – 64 Averages

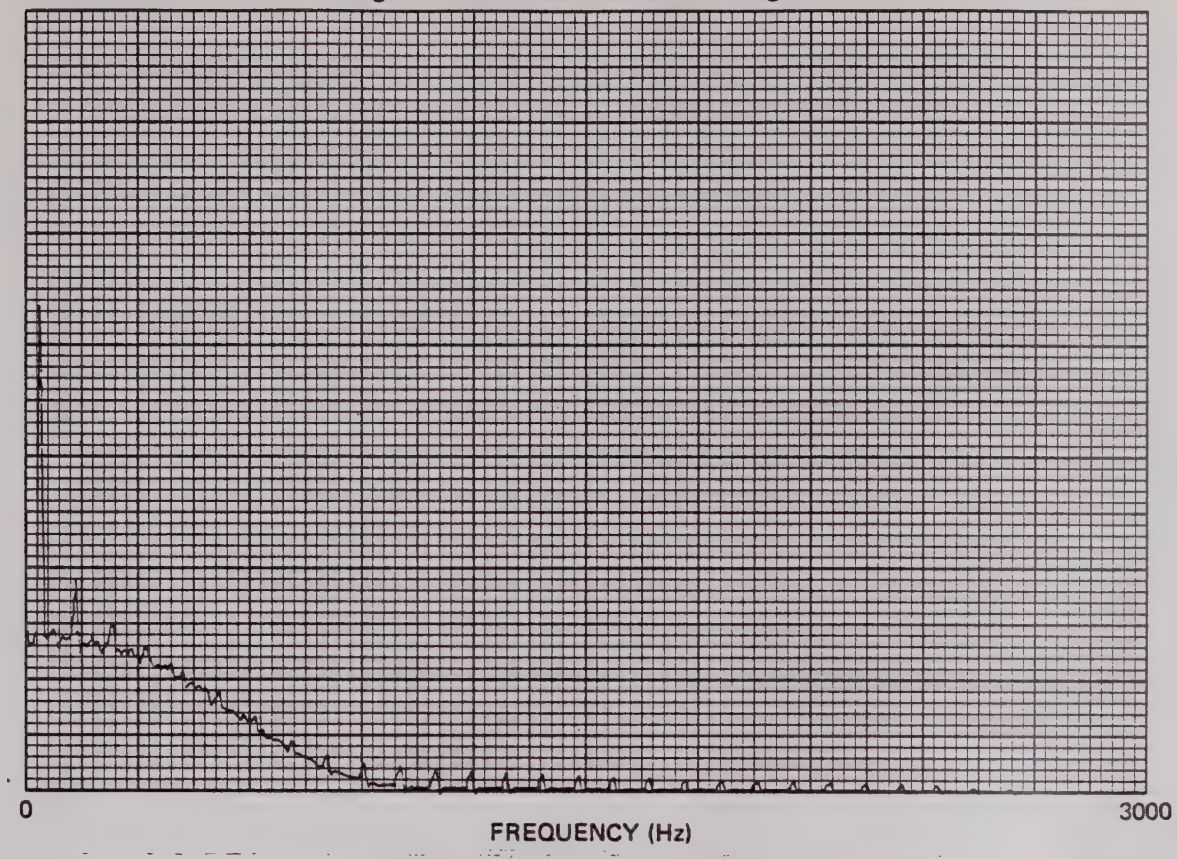
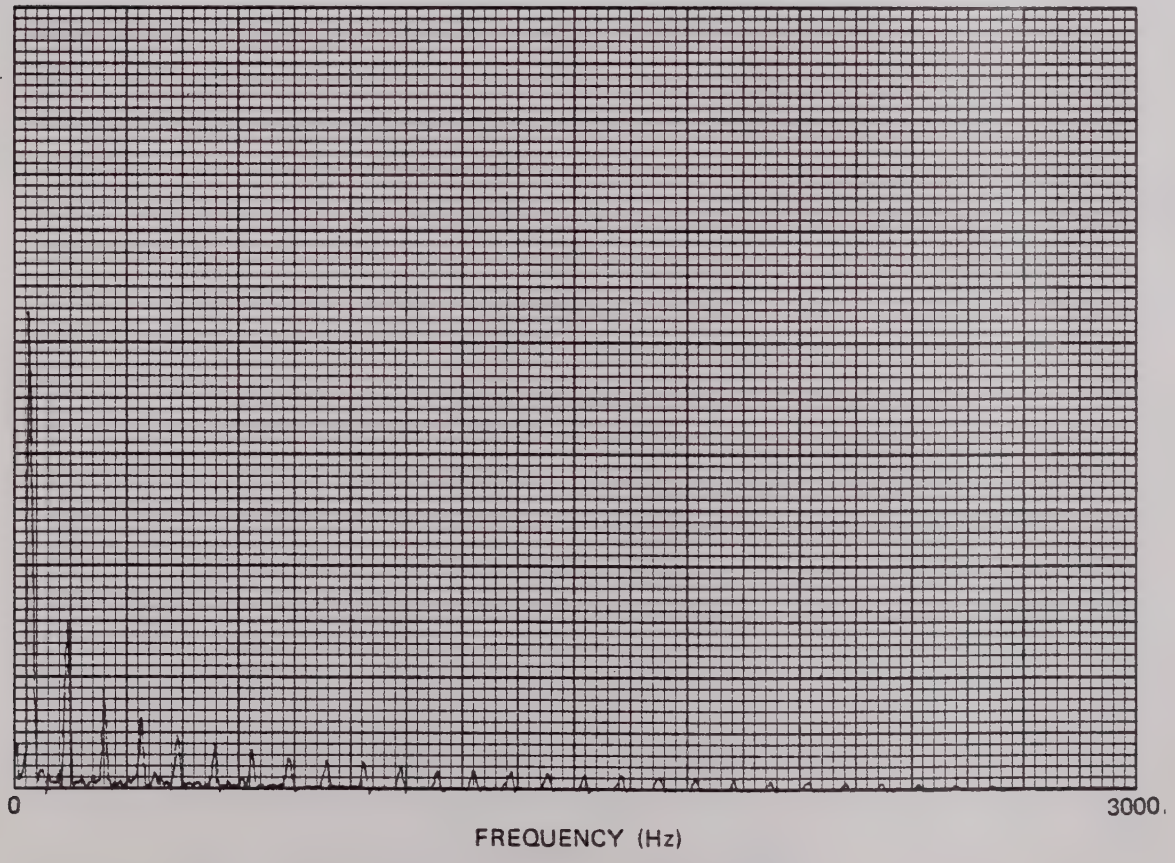


Figure 87. Function 13 – 64 Averages



FUNCTION 14 – PROBABILITY DENSITY HISTOGRAM

Probability density is the relative probability of occurence of a signal level within a given amplitude window at any instant of time. The probability density histogram is independent of frequency range of the signal. Amplitude scaling is very important, however, as the X axis of the display is now amplitude.

The 12-bit ADC has 4096 digitizing levels which are divided into 128 class intervals (Δx windows) of 32 digitizing levels each for performing function 14. The zero amplitude class interval begins in output memory location 512. Input signals are sampled by the ADC and when any one of the 32 levels of a class are detected, a 1 is placed in the output memory location for that class interval, where they are accumulated. The input memory has 1024 words; thus, the maximum values of any p(x) for a time ensemble is 1024.

The probability density histogram display has the zero amplitude class interval beginning at the center of the X axis with 64 Δx windows on each side. For display the processor addresses the memory location for each amplitude class interval eight consecutive times, producing a display of discrete steps eight "cells" wide, representing the relative probability of occurrence of the input signal at all scaled amplitudes.

Function 14 is done with LIN averaging, DO NOT perform more than 1024 averages.

$$p(x_j) = \sum_{i=0}^{1023} v(x_i) \text{ where } v(x_i) \begin{cases} 1 \text{ if } x_j \leq x_i < x_j + \Delta x \\ 0 \text{ if otherwise} \end{cases}$$

$p(x_j)$ = The number of times that the sampled values x_i fall within the digitizing window x_j to $x_j + \Delta x$

i = sample number

j = 1, 2, 3, ..., 128 amplitude window index

Figure 88 is the data flow for function 14.

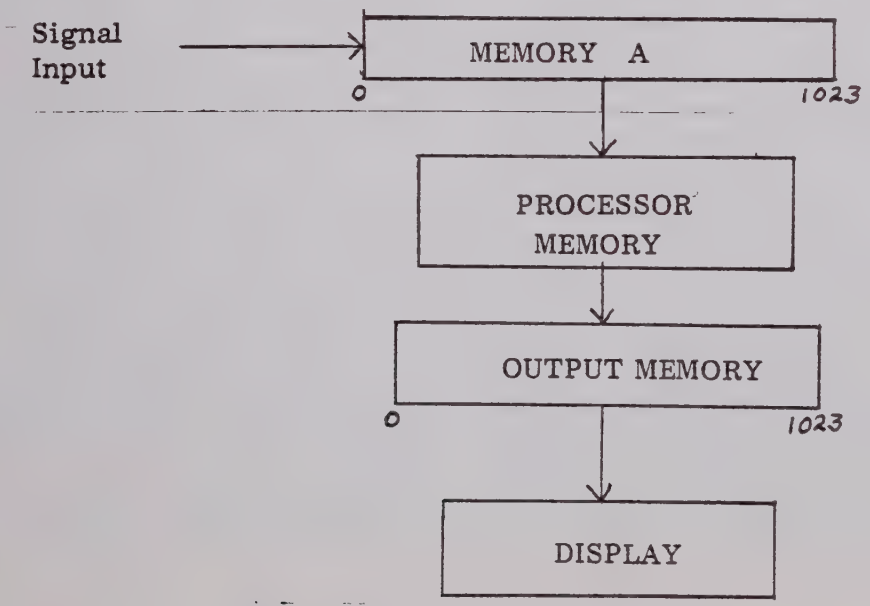
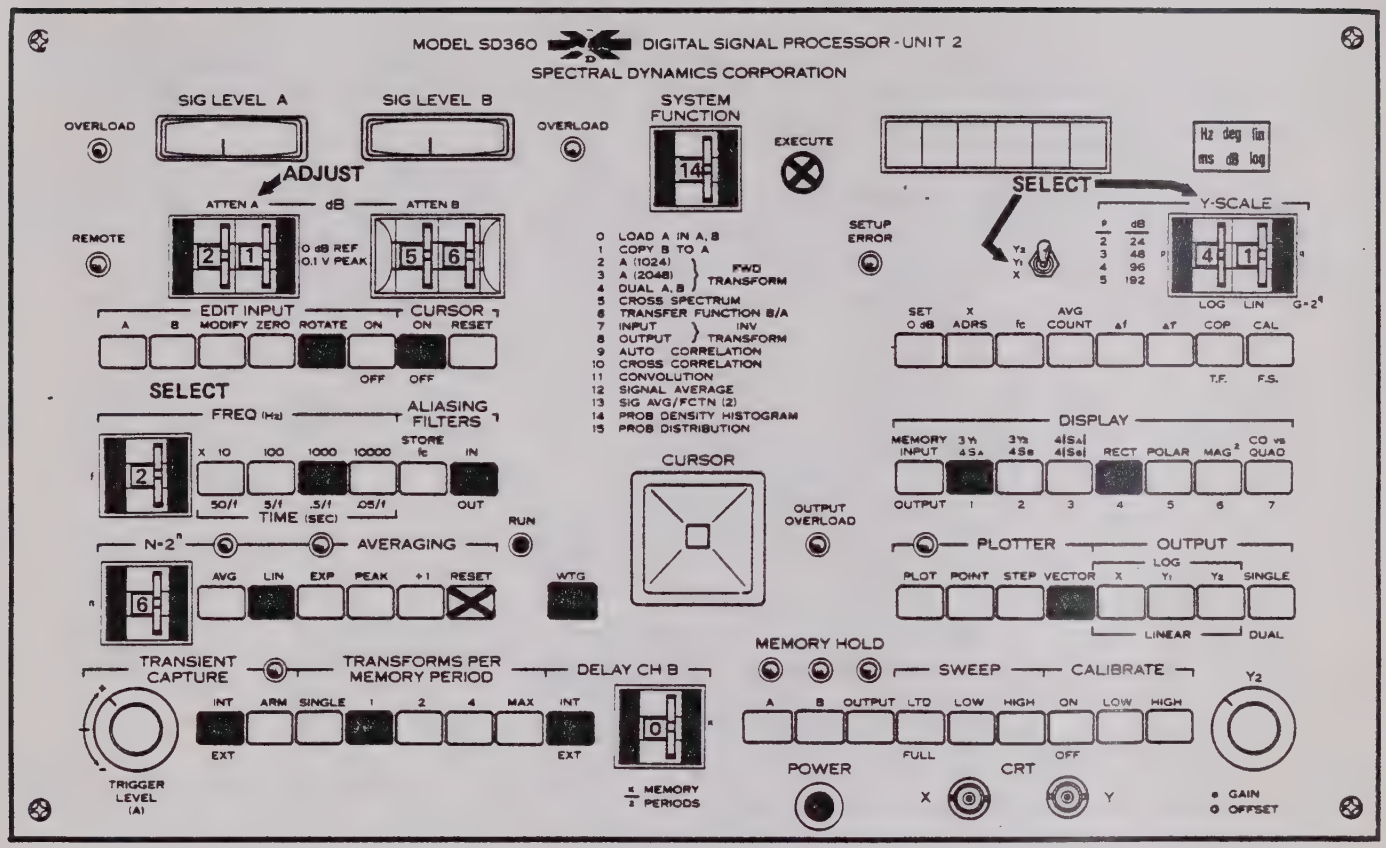


Figure 88. Data Flow For Function 14

Figure 89. Function 14 Diagram



FUNCTION 15 – PROBABILITY DISTRIBUTION

The probability distribution gives the probability that a given amplitude level is not exceeded. It is equal to the area under the probability density curve, or for discrete values, the sum of the probability density histogram values below the given amplitude level.

A Function 14 must be executed and the probability density histogram (PDH) data stored in output memory before performing Function 15. As the PDH data is read out of memory, the probability distribution is computed. Figures 91-94 are examples of results of performing Functions 14 and 15.

Probability Distribution Function

$$P(x_k) = \sum_{\ell=0}^{k} p(x_\ell)$$

$\ell = 0, 1, 2, \ldots., 1023$ memory address pointer

$k = 0, 1, 2, \ldots., 1023$

Figure 90. Function 15 Diagram

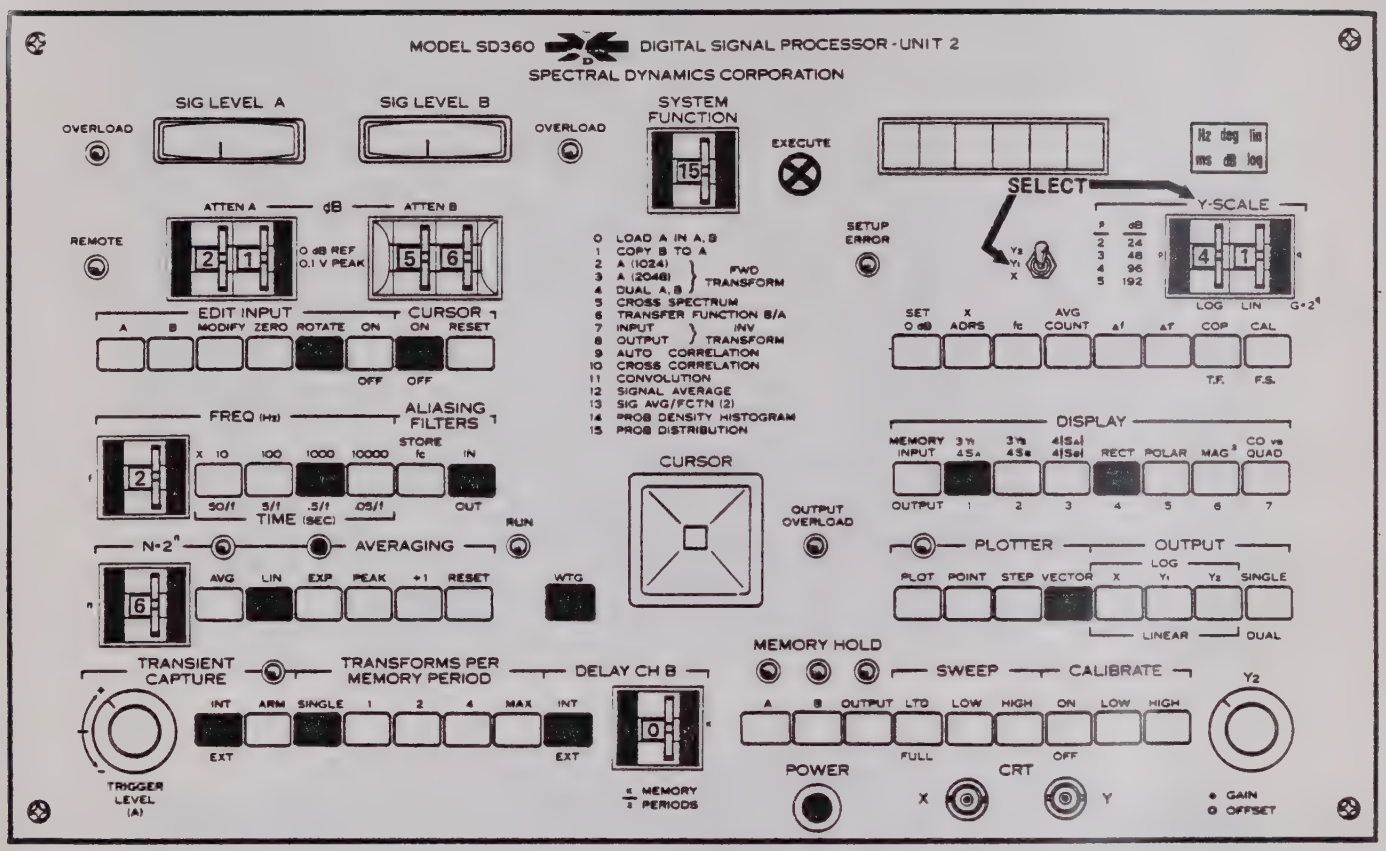


Figure 91. Function 14 — Probability Density Histogram-Sine Wave

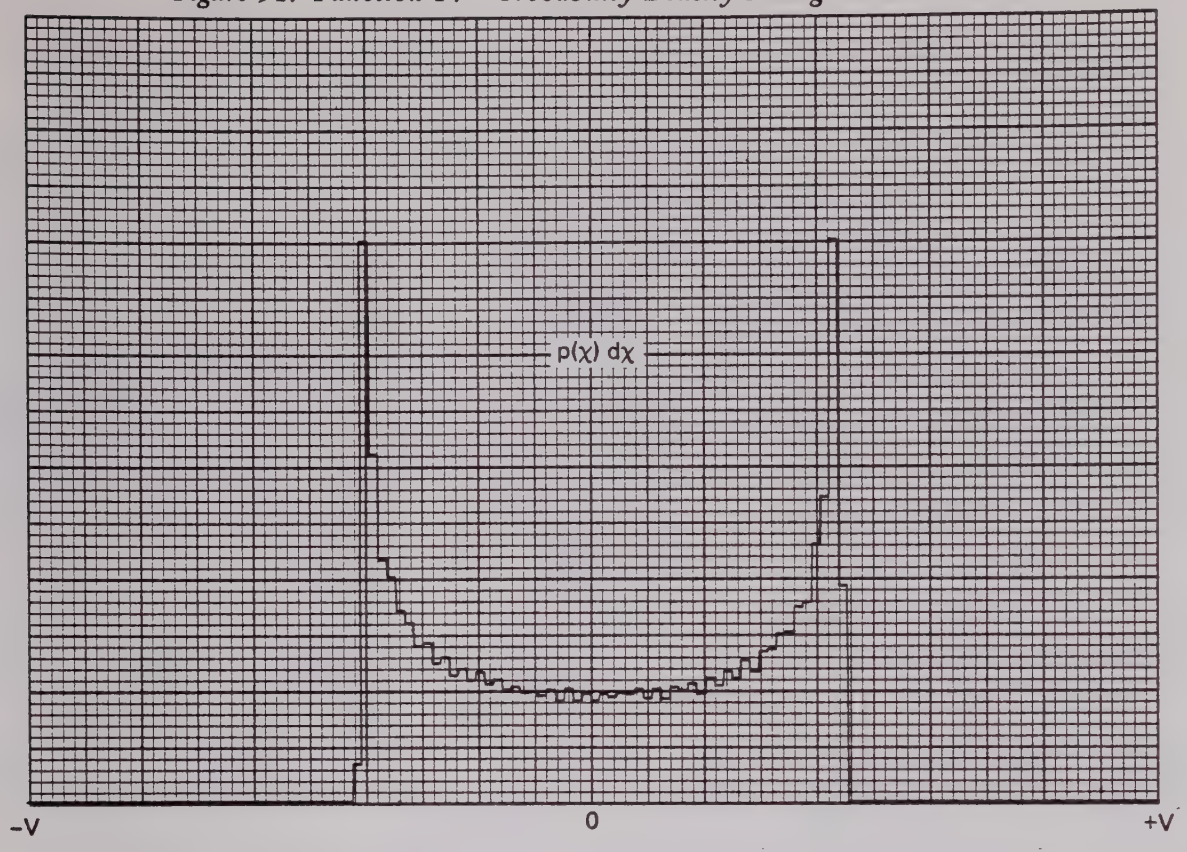


Figure 92. Function 15 — Probability Distribution-Sine Wave

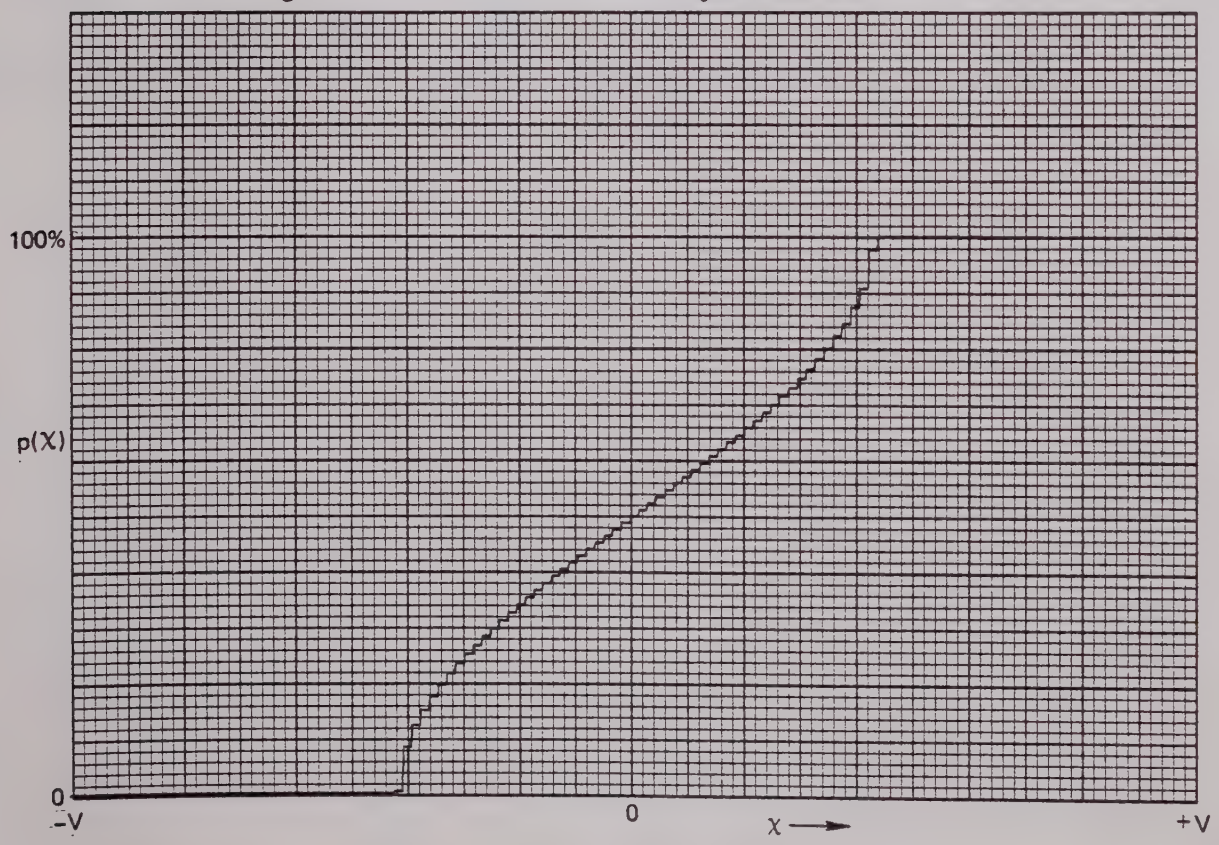


Figure 93. Function 14 – Probability Density Histogram-Gaussian Noise

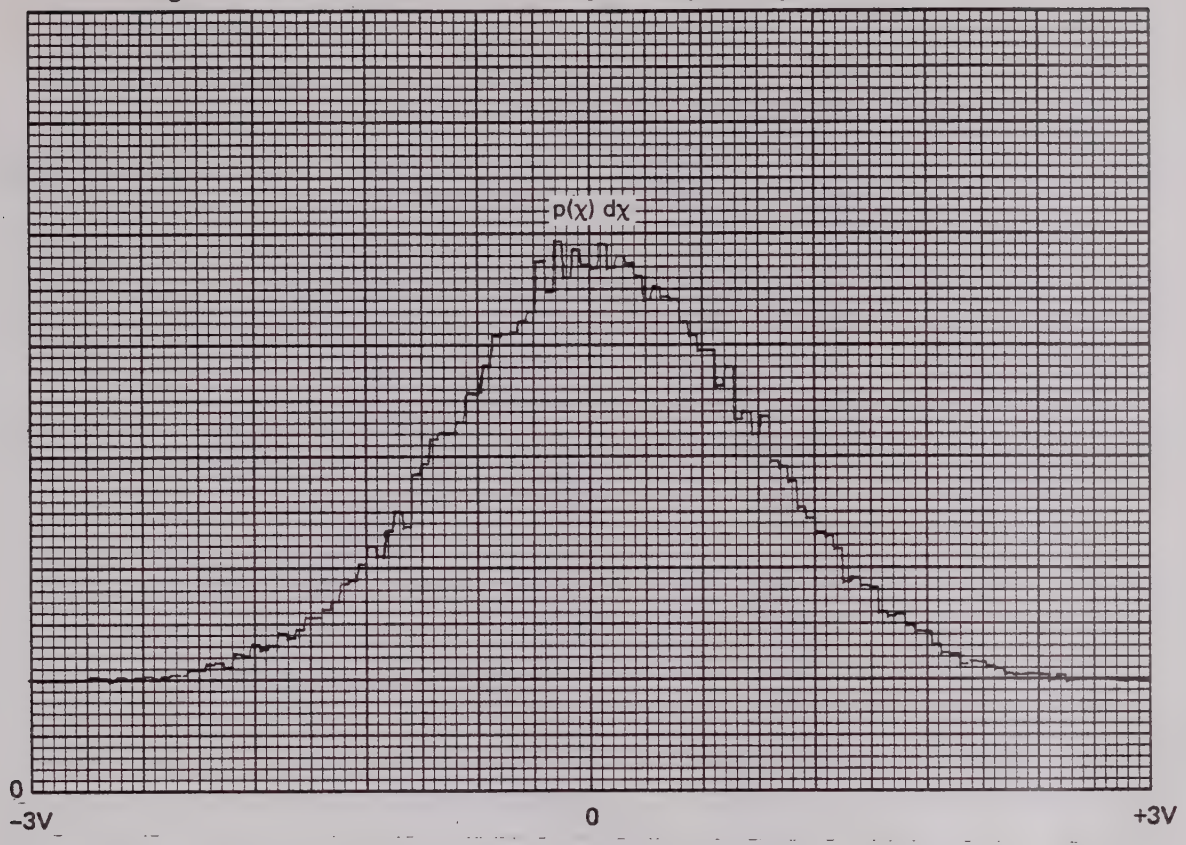
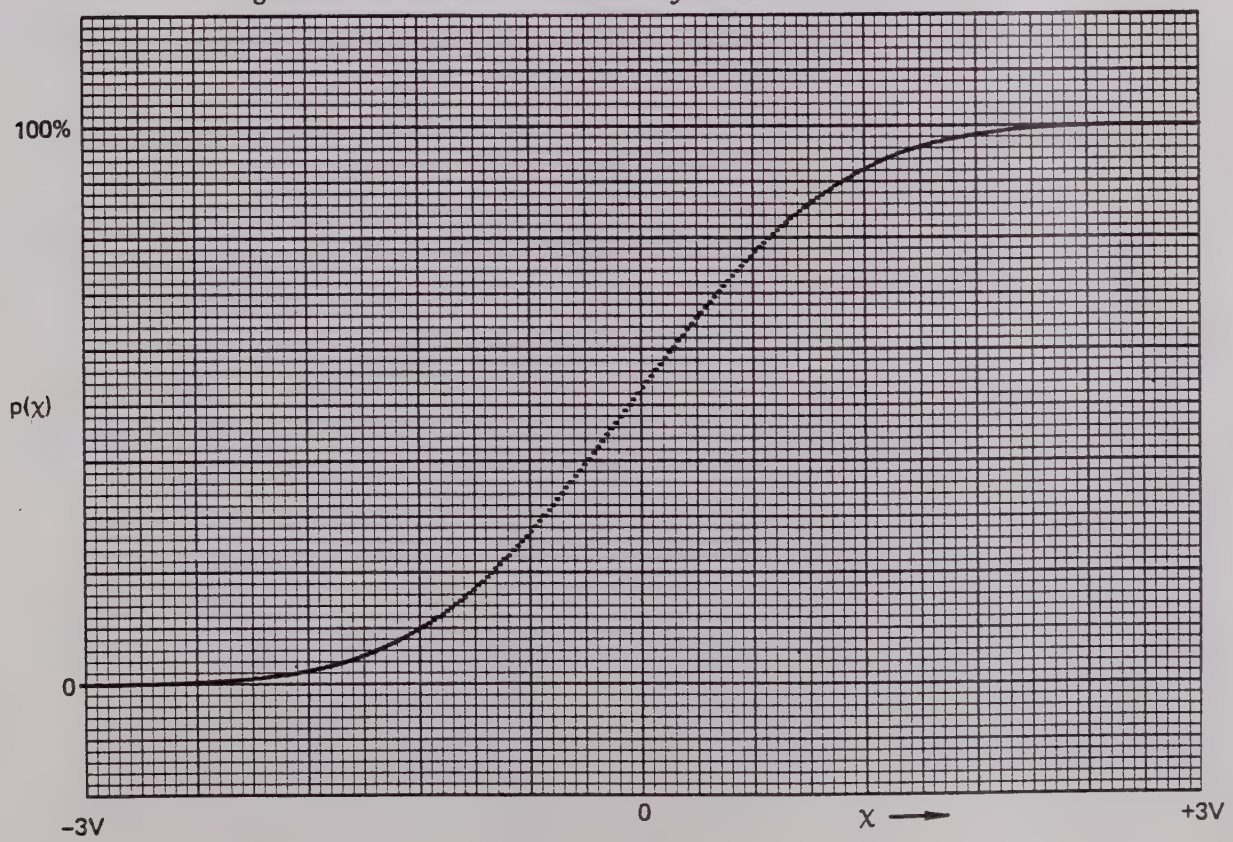


Figure 94. Function 15 – Probability Distribution-Gaussian Noise



APPENDIX A

INTERPRETING SOME COMMON PROBABILITY DENSITY FUNCTIONS

When a probability density function is performed on the SD360, the result has the abscissa given in terms of instantaneous voltage (x) and the ordinate expressed in terms of relative probability of occurrence p(x). Ideally, the total area under the probability density curve is 1, indicating that the data signal always occurred somewhere within the bounded measurement. In addition the probability density function implies a division by the analysis window, Δx, so that the probability density function is often expressed as p(x)dx. To calculate the probability of occurrence of a signal within a given range of amplitudes, Δx_1, multiply the probability density function over that range by the selected window width. This gives the product $px(dx)\,\Delta x_1$ which is the probability of the signal occurring within that window. Note that the value of the probability density function at any one point is meaningless until a selected window width, no matter how small, is associated with it.

To simplify the presentation and interpretation of the display even more, the display is often normalized along the abscissa by dividing by the signal's true rms level, σ. Often the display will be p(x)dx vs $\frac{x}{\sigma}$. The display would then be carried out to ±3 or 4 σ. This necessitates having a knowledge of the running rms level of the signal input, which is not calculated by the SD360. Through a careful calibration procedure, the probability density function for a stationary signal can be displayed on a normalized abscissa basis.

CALIBRATION FOR A RANDOM SIGNAL INPUT

A stationary random signal which follows a Gaussian distribution can be described by the expression

$$p(x) = \frac{1}{\sigma\sqrt{2\pi}}\,\epsilon^{-(x-x_0)^2 \Big/ 2\sigma^2} \qquad (1)$$

where x = the instantaneous signal amplitude

x_0 = the mean value of the signal

σ = the true rms value of the signal

σ^2 = the variance of the random signal

Here the parameters of the signal have been "normalized" to assure that

$$\int_{-\infty}^{+\infty} p(x)dx = 1 \qquad \text{and also} \qquad (2)$$

$$m_1 \text{ (the first moment)} = \int_{-\infty}^{+\infty} x\,p(x)dx = x_0 \qquad (3)$$

$$m_2 \text{ (the second moment)} = \int_{-\infty}^{+\infty} x^2\,p(x)dx = x_0^2+\sigma^2 \qquad (4)$$

$$\mu_2 \text{ (the variance} = \int_{-\infty}^{+\infty} (x-m_1)^2\,p(x)dx = m_2-m_1^{\,2} \qquad (5)$$

$$= \sigma^2$$

For ac coupled data the mean value, x_0, will be zero. We can then calculate the value of the probability density function for $x = x_0 = 0$

$$p(0) = \frac{1}{\sigma\sqrt{2\pi}}$$

Assume $\sigma = 1$, then p(0) = 0.399 which gives a readily accepted calibration point for the vertical display.

CALIBRATION FOR A SINUSOIDAL SIGNAL INPUT

Strictly speaking, since a sinusoid is periodic and deterministic, there is nothing probablistic about it and determining its probability density function is sheer folly! Nevertheless, often calibration signals are sinusoidal; and therefore, the numerical value of the resulting density function should be examined. However, before this can be done, several concepts in functional relationships must be introduced.

In general, given the probability density function, $q(\theta)$, of a variable, θ, to calculate the probability

density function p(x) for $x = f(\theta)$ it is convenient to invert the functional relationship and write $\theta = f(x)$. Then, by direct substitution

$$q(\theta)d\theta = q\left[f(x)\right] f'(x)dx \qquad (6)$$

$$\text{where } f'(x) = \frac{d\left[f(x)\right]}{dx}$$

Hence, if $f'(x)$ is a single valued function of x

$$p(x) = q\left[f(x)\right] \left|f'(x)\right| \qquad (7)$$

If $f(x)$ is multiple valued, the expression on the right is summed over all the values. For the sinusoidal case, let $x = A \sin\theta$ where θ has a uniform distribution in the range $-\pi/2$ to $+3\pi/2$. Then the values of x are distributed like those of the ordinates of a sine wave with amplitude, A. To find the probability density function for x,

$$q(\theta) = \begin{cases} 0 & \frac{3\pi}{2} < \theta < \frac{-\pi}{2} \\ \frac{1}{2\pi} & \frac{-\pi}{2} < \theta < \frac{3\pi}{2} \end{cases}$$

where $\frac{1}{2\pi}$ insures a normalized condition.

$$\text{Since } \sin\theta = \frac{x}{A} \qquad (8)$$

$$\text{then } \theta = f(x) = \arcsin\frac{x}{A}$$

$$f'(x) = \frac{d}{dx}\left[f(x)\right] = \left[A^2 - X^2\right]^{-\frac{1}{2}} \qquad (9)$$

Recall that

$$p(x) = q\left[f(x)\right] \left|f'(x)\right|$$

and that each value of x in the range $-A$ to A corresponds to two values of θ, multiply the right side of the above equation by 2 to obtain p(x).

$$p(x) = \frac{1}{\pi\sqrt{A^2 - X^2}} \qquad -A < X < A$$

$$p(x) = 0 \qquad \text{otherwise}$$

Two features of this function seen at once are:

The density function grows without bound at $x = \pm A$.

For the sinusoidal case $A = 1.414\,\sigma$ so that for $x = 0$,

$$p(x) = \frac{1}{\pi\sqrt{(1.414\sigma)^2}} = \frac{1}{4.4422\sigma} = \frac{0.22511}{\sigma}$$

If again $\sigma = 1$, the value of the probability density function at $x = 0$ is 0.22511. This point can also be used as a convenient calibration value. The probability density and distribution functions for a sine signal are given in Figure A-1.

USE OF A TRIANGULAR WAVEFORM FOR SYSTEM CALIBRATION

The triangular waveform offers a convenient check for probability calculations since it has a constant probability value. Stated another way, the triangular waveform has an equal chance of being anywhere within its complete cycle when observed over any time increment which is small with respect to a complete period. With this in mind it is a simple manner to compute the triangular density function. Assuming a zero mean value, the symmetrical density function extends from $-A$ to $+A$ with the total area under the curve equal to 1. Thus $2A\,p(x) = 1$.

For a triangular waveform, the peak value, A, is equal to $\sqrt{3}$ times the rms level or $A = \sqrt{3}\,\sigma$. Thus, $2\sqrt{3}\,\sigma\,p(x) = 1$, or

$$p(x) = \frac{1}{2\sqrt{3}\,\sigma} = \frac{0.288675}{\sigma} \qquad (10)$$

Again let $\sigma = 1$, which gives a convenient calibration point for any place on the waveform. The probability density and distribution functions for a triangular waveform are seen in Figure A-2.

EXPRESSING THE CUMULATIVE DISTRIBUTION FUNCTION

Another function of importance is the cumulative probability distribution function. Total probability is represented by the area under the probability density curve which must equal 1. The probability distribution function represents the running integra-

tion or area under the curve of the density function. Traditionally, the distribution function is integrated from the lowest to the highest abscissa value giving it a value from 0 to 100%. The distribution function can never decrease when integrating from the lower to the upper limit of the density function. It is monotonic although it may have one or more inflection points.

The distribution function is defined as the probability that a value is less than some specified x;

$$P(x_i) = \int_{-\infty}^{x_i} p(x)dx \qquad (11)$$

$$P(-\infty) = 0 \text{ and } P(+\infty) = 1.$$

For Gaussian noise, the distribution function can be expressed as

$$P\left(\frac{|x|}{\sigma} \geqslant\right) = \frac{2}{\sqrt{2\pi}} \int_{x_1/\sigma}^{\infty} \epsilon^{-(x_1^2/2\sigma^2)}\, d\left(\frac{x_1}{\sigma}\right) \qquad (12)$$

This is the probability of a signal being "greater than" some arbitrary value. It can also be expressed as before as the probability of a signal being "less than" some value.

If a probability density function is symmetrical about some mean value and the mean is zero, then the probability that x exceeds some given absolute value, $|x|$, is twice the probability that it exceeds the same absolute value in either the negative or positive sense. Therefore, it is sometimes convenient to plot the distribution function as $P\left(\frac{|x|}{\sigma} \geqslant\right)$ in terms of the absolute value.

For the sinusoidal case, the distribution function can also be described several ways. For example, the probability of being less than some arbitrary value can be expressed as

$$P(x) = \int_{-A}^{x} \frac{dx}{\pi\sqrt{A^2 - X^2}} \qquad (13)$$

$$= \frac{1}{\pi}\left(\frac{\pi}{2} + \arcsin\frac{x}{A}\right)$$

Or in terms of the "absolute" probability of being greater than some value

$$P\left(\frac{|x|}{\sigma} \geqslant\right) = \frac{2}{\pi} \arccos \frac{x}{\sigma\sqrt{2}} \qquad (14)$$

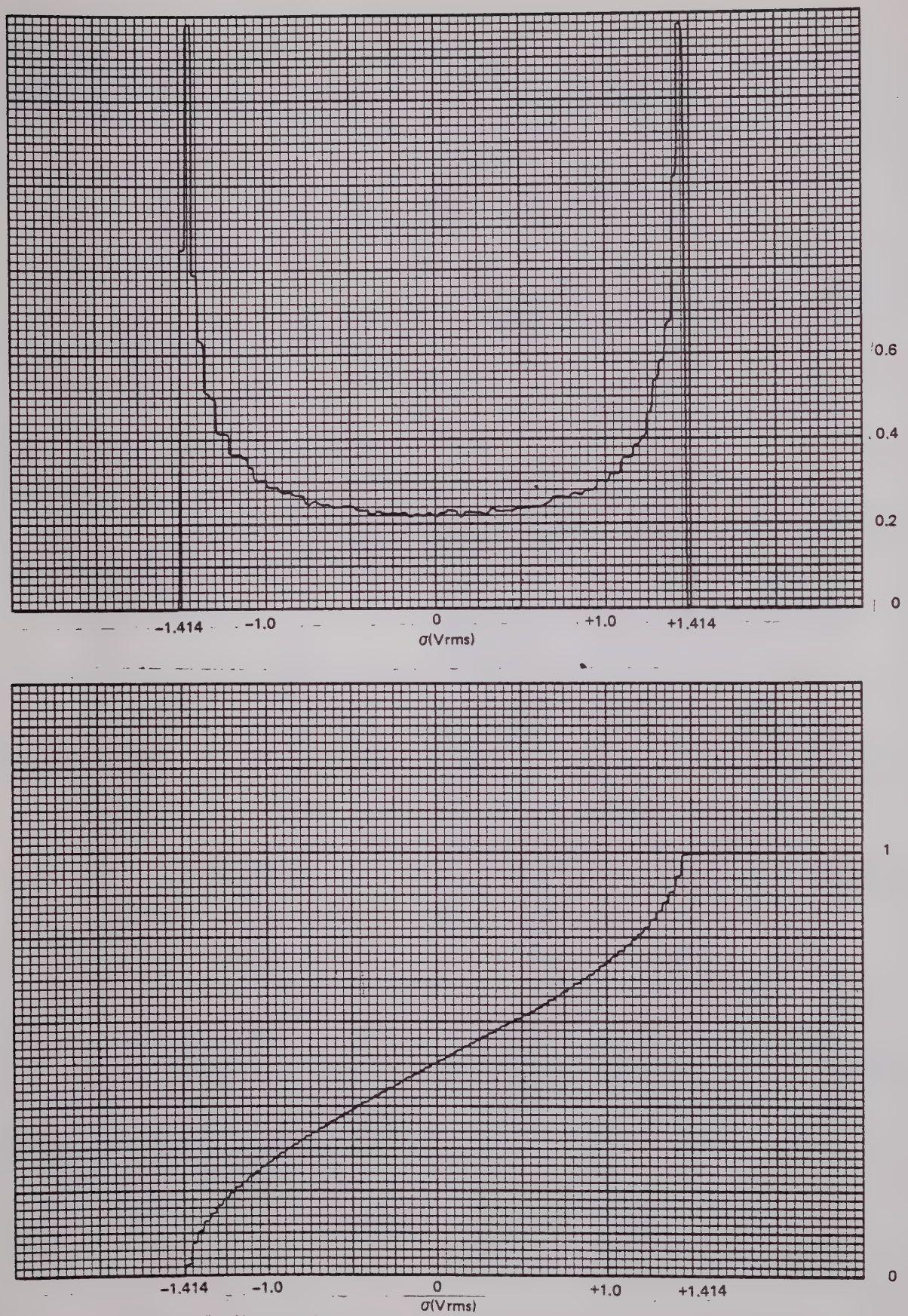
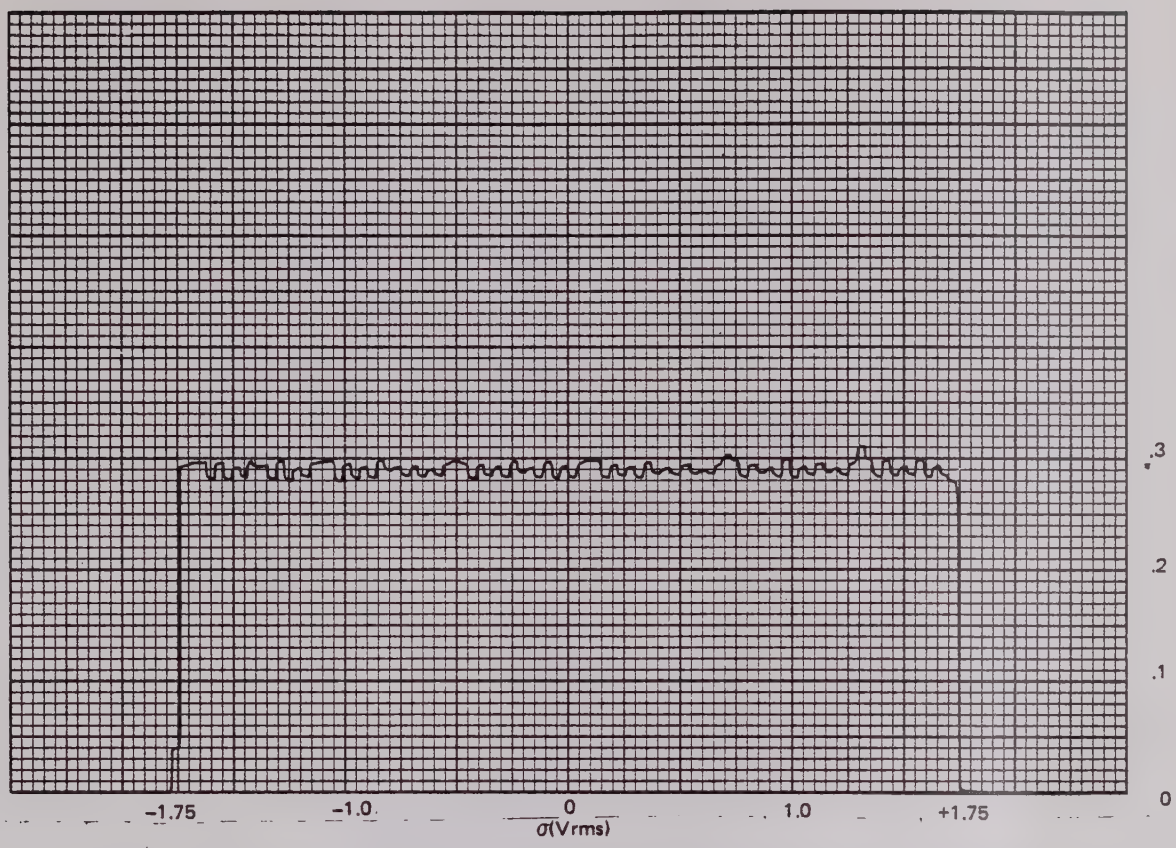


Figure A-1. Probability Density and Distribution Functions, Sinusoidal Input



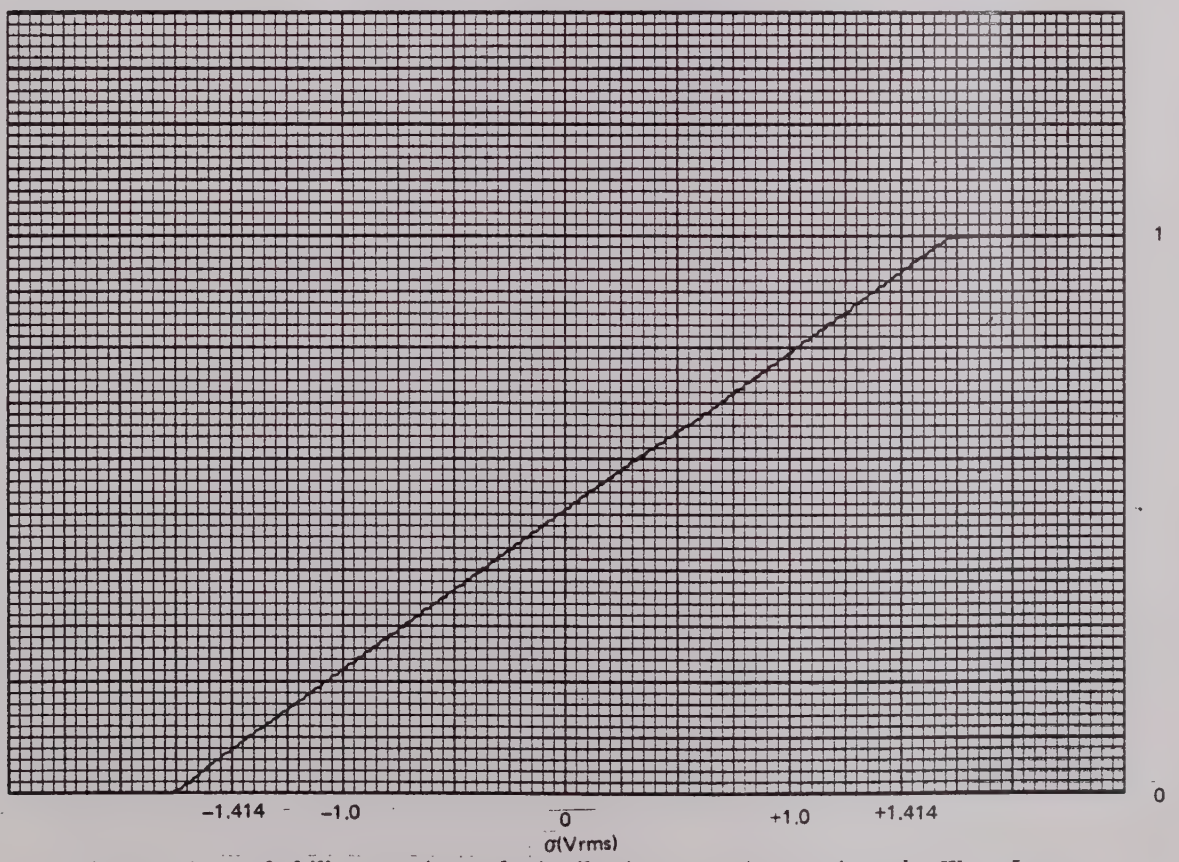


Figure A-2. Probability Density and Distribution Functions, Triangular Wave Input

APPENDIX B

INTERPRETING CORRELATION FUNCTIONS

The Model SD360 Digital Signal Processor computes the cross-correlation function by first computing the cross-spectrum and then calculating an inverse Fourier transform via the cross-correlation function. The cross-correlation function is calculated for two different data signals. The autocorrelation function, as a special case of this, is computed in the same general fashion by taking the inverse Fourier transform of the power spectrum which was first computed from a single input source. The mathematical development of these functions is as shown below:

As a direct correlation function,

$$R_{xx}(\tau) = \lim_{T \to \infty} \frac{1}{T} \int_0^T f(t)\, f(t+\tau)\, dt \qquad (1)$$

This states that multiplying a time signal by delayed replica of itself and integrating the produ over a suitable time frame versus time delay pr duces a normalized autocorrelation function.

Taking the same time function, $f(t)$, its Fouri transform is

$$G(j\omega) = \int_{-\infty}^{+\infty} f(t)\, \epsilon^{-j\omega t}\, dt \qquad (2)$$

Doing an inverse Fourier transform

$$f(t) = \frac{1}{2\pi} \int_{+\infty}^{-\infty} G(j\omega)\, \epsilon^{j\omega t}\, d\omega \qquad (3)$$

the original time signal is recovered. Equations and 3 form a Fourier transform pair.

The autocorrelation function can also be expresse in terms of a signal's power spectrum.

Thus, if equation 2 is the Fourier transform of $f(t$ its power spectrum is

$$G_{xx}(\omega) = \lim_{T \to \infty} \frac{1}{T} \overline{|G(j\omega)|}^2 \qquad (4)$$

This is, strictly speaking, a two-sided power spectrum extending over positive and negative ω.

The Wiener-Khintchine theorem states that

$$R_{xx}(\tau) = \frac{1}{2\pi} \int_{-\infty}^{+\infty} G_{xx}(j\omega)\, \epsilon^{j\omega\tau}\, d\omega \qquad (5)$$

or that the autocorrelation function is the inverse Fourier transform of the power spectrum.

Similarly,

4. If $R_{xx}(\tau) = \frac{1}{2T} \int_0^1 f_1(t)\, f_1(t+\tau)\, dt$

then $R_{xx}(0)$ = mean square value of $f_1(t)$.

If $f_1(t)$ is a voltage or current, $R_{xx}(0)$ is just the average power dissipated in a one ohm resistor.

5. $R_{xx}(\tau)$ is less than or equal to $R_{xx}(0)$;

6. If the input signal $S_1(t)$ contains periodic components (or a d-c value), the autocorrelation function contains components of the same period (or a d-c component);

7. For a random signal with no periodic or d-c components, the limit of $R_{xx}(\tau)$ as τ approaches infinity equals 0;

8. $R_{xx}(\tau)$ is continuous everywhere if it's continuous at the origin;

9. $R_{xx}(\tau)$ and $G_{xx}(\omega)$ (the power spectral density) are Fourier transforms one of the other;

10. It is necessary and sufficient for a function $\xi(t)$ to be an autocorrelation function, that it be real and even, have a positive and finite value at the origin, and have a non-negative Fourier transform;

11. The autocorrelation function of a sum of signals (sinusoidal or otherwise) is just the sum of the autocorrelation functions of the individual signal components;

12. A given $R_{xx}(\tau)$ corresponds to an infinite number of time functions (it is essentially phase independent), but each time function has a unique $R_{xx}(\tau)$;

13. $$R'_{xx}(\tau) = \frac{d}{dt}\left[R_{xx}(\tau)\right] = \lim_{T\to\infty} \frac{1}{2T} \int_{-T}^{+T} f_1(t) f_1'(t+\tau)\,dt \qquad (7)$$

14. $R''_{xx}(\tau) = -R_{xx}(\tau)$ where $R_{xx}(\tau)$ is the autocorrelation of $f_1'(t)$

15. If the function $f_1(t)$ contains a noise component unrelated to any desirable signal in $f_1(t)$, then $R_{xx}(\tau)$ will contain a noise component which will tend toward 0 as t approaches infinity.

Similarly, any interpretation of a processed cross-correlation function should be based on at least the following properties of $R_{xy}(\tau)$:

1. $R_{xy}(\tau)$ is not necessarily an even function; in fact, $R_{xy}(-\tau) = R_{yx}(\tau)$;

2. The value of $R_{xy}(\tau)$ at $\tau = 0$ is not necessarily the maximum;

3. $R_{xy}(\tau)$ is not phase independent. If one of the input signal functions, $f_1(t)$ or $f_2(t)$, is moved T_1 units, then $R_{xy}(\tau)$ will be moved T_1 units;

4. Noise does not add to the cross-correlation function between two periodic functions of the same frequency. As a matter of fact, the cross-correlation function of a noisy signal with one containing no noise contains no noise.

APPENDIX C

The Basic FFT Concept

To implement the Fourier transform for automatic computation and to take advantage of high speed computing machinery, a digital representation of the continuous waveforms which appear in nature is necessary. This is accomplished by time sampling a continuous waveform at fixed, discrete intervals. A series of such samples can then be processed for their Fourier coefficients using

$$H(k) = \sum_{n=0}^{N-1} h(n)\, e^{-j\frac{2\pi}{N}nk} \qquad (1)$$

The Fourier coefficients, H(k), have been calculated from the discrete sampled values of the waveform, h(n). The term $e^{-j\frac{2\pi}{N}}$ may be factored out as W, giving

$$H(k) = \sum_{n=0}^{N-1} h(n)\, W^{nk} \qquad (2)$$

To recover an original waveform (or actually the sampled values of the waveform), h(n), the expression

$$h(n) = \sum_{k=0}^{N-1} H(k)\, e^{j\frac{2\pi}{N}nk} \qquad (3)$$

is used. Equations 1 and 3 form a discrete Fourier transform pair.

Examining Equation 1 carefully, certain parameters become apparent. For example, it tells us that for each sampled data value input, a summation is formed of the cross products between the original data sample and each of the sine/cosine values represented by one complete circulation of the rotating parameter $e^{-j\omega t}$. Assume that 1024 original data samples are taken. For reasons of symmetry it is helpful if the number of data samples taken corresponds to an exact power of 2. Since the complex exponential is also going to be broken up into 1024 possible states or values, each data sample will cause 1024 complex operations. Since we have 1024 data samples, the total number of complex operations is nominally 1024^2, or over 1,000,000 operations. One million operations take an appreciable amount of time, which would almost preclude any type of "real time" operation. For this reason, a much faster processing algorithm was developed, the Fast Fourier Transform.

The discrete Fourier transform can be described by (Ref. 1),

$$H(k) = \sum_{n=0}^{N-1} h(n)\, e^{-j\frac{2\pi}{N}nk} \qquad k = 0, 1, 2, \ldots N-1 \tag{4}$$

where

$$e^{-j\frac{2\pi}{N}nk} = W \tag{5}$$

If the total number of real data points (N) is 4, the summation can be easily represented in Matrix form as

$$\begin{bmatrix} H(0) \\ H(1) \\ H(2) \\ H(3) \end{bmatrix} = \begin{bmatrix} W^0 & W^0 & W^0 & W^0 \\ W^0 & W^1 & W^2 & W^3 \\ W^0 & W^2 & W^4 & W^6 \\ W^0 & W^3 & W^6 & W^9 \end{bmatrix} \begin{bmatrix} h_0(0) \\ h_0(1) \\ h_0(2) \\ h_0(3) \end{bmatrix} \tag{6}$$

In a more compact form, this can be written as

$$H(k) = W^{nk} h(n) \tag{7}$$

An examination of Equation 6 shows that since W and possibly $h_0(k)$ are complex, then N^2 complex multiplications and $N(N-1)$ complex additions are necessary to perform the required matrix computation. The FFT owes its success to the fact that its algorithm reduces the number of multiplications and additions required in the computation of Equation 6.

Applying the FFT concept to Equation 6 and rewriting, we obtain

$$\begin{bmatrix} H(0) \\ H(1) \\ H(2) \\ H(3) \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & W^1 & W^2 & W^3 \\ 1 & W^2 & W^0 & W^2 \\ 1 & W^3 & W^2 & W^1 \end{bmatrix} \begin{bmatrix} h_0(0) \\ h_0(1) \\ h_0(2) \\ h_0(3) \end{bmatrix} \tag{8}$$

Next, factor 8 as follows.

$$\begin{bmatrix} H(0) \\ H(2) \\ H(1) \\ H(3) \end{bmatrix} = \begin{bmatrix} 1 & W^0 & 0 & 0 \\ 1 & W^2 & 0 & 0 \\ 0 & 0 & 1 & W^1 \\ 0 & 0 & 1 & W^3 \end{bmatrix} \begin{bmatrix} 1 & 0 & W^0 & 0 \\ 0 & 1 & 0 & W^0 \\ 1 & 0 & W^2 & 0 \\ 0 & 1 & 0 & W^2 \end{bmatrix} \begin{bmatrix} h_0(0) \\ h_0(1) \\ h_0(2) \\ h_0(3) \end{bmatrix} \tag{9}$$

This factorization is the key to efficiency of the FFT algorithm. If the two square matrices of 9 are multiplied, they yield the square matrix of 8 with the exception that rows 1 and 2 will have been interchanged. This interchange was taken into account in Equation 9 by rewriting the column vector H(k). This row interchanged vector will be denoted by $\overline{H(k)}$.

Note that the quantity, $e^{-j\frac{2\pi}{N}}$, is a point on the unit circle and W^n is also a point on the circle with an angle n times the size of the angle associated with W; that is, $n \cdot 2\pi/N$. W is periodic in N terms so that there are only N distinct points W^n. Half of the points are negative of some other point on the circle. For example, with a 16-point transform, $W^{10} = -W^2$.

Write (10)

$$\begin{bmatrix} h_1(0) \\ h_1(1) \\ h_1(2) \\ h_1(3) \end{bmatrix} = \begin{bmatrix} 1 & 0 & W^0 & 0 \\ 0 & 1 & 0 & W^0 \\ 1 & 0 & W^2 & 0 \\ 0 & 1 & 0 & W^2 \end{bmatrix} \begin{bmatrix} h_0(0) \\ h_0(1) \\ h_0(2) \\ h_0(3) \end{bmatrix}$$

Recall that the column vector $h_1(n)$ is equal to the product of the two matrices on the right of Equation 9. The first element of Equation 10 is computed by one complex multiplication and one complex addition as follows. (11)

$$h_1(0) = h_0(0) + W^0 h_0(2)$$

The next element of 10, that is $h_1(1)$, is also determined by one complex multiplication and one complex addition. Only one complex addition is required to compute $h_1(2)$. This follows from the fact that $W^0 = -W^2$. Hence, (12)

$$\begin{aligned} h_1(2) &= h_0(0) + W^2 h_0(2) \\ &= h_0(0) - W^0 h_0(2) \end{aligned}$$

where the complex multiplication $W^0 h_0(2)$ has already been computed in the determination of $h_1(0)$ as Equation 11. Using similar reasoning for $h_1(3)$, we see that the intermediate vector $h_1(k)$ has been determined by four complex additions and two complex multiplications. We then complete the computation of 9 using our just calculated intermediate vector by writing (13)

$$\begin{bmatrix} H(0) \\ H(2) \\ H(1) \\ H(3) \end{bmatrix} = \begin{bmatrix} h_2(0) \\ h_2(1) \\ h_2(2) \\ h_2(3) \end{bmatrix} = \begin{bmatrix} 1 & W^0 & 0 & 0 \\ 1 & W^2 & 0 & 0 \\ 0 & 0 & 1 & W^1 \\ 0 & 0 & 1 & W^3 \end{bmatrix} \begin{bmatrix} h_1(0) \\ h_1(1) \\ h_1(2) \\ h_1(3) \end{bmatrix}$$

The term $h_2(0)$ is determined by one complex multiplication and one complex addition. (14)

$$h_2(0) = h_1(0) + W^0 h_1(1)$$

Computation of H(k) by means of Equation 9 required a total of four complex multiplications and eight complex additions. Computation of H(k) by Equation 6 would have required 16 complex multiplications and 12 complex additions. Note

that the matrix factorization process introduces zeros into the factored matrices and, as a result, reduces the required number of multiplications. For this example, the matrix factorization process reduced the number of multiplications by a factor of two. Since computation time is largely governed by the required number of multiplications, it is obvious that the FFT algorithm can be quite efficient.

For $N = 2^p$ the FFT algorithm is simply a procedure for factoring an N x N matrix into matrices (each N x N) such that each of the factored matrices has the special property of minimizing the number of complex multiplications and additions. Extending the result of the previous example, the FFT requires $Np/2 = 4$ complex multiplications and $N = 8$ complex additions; whereas the direct method (Equation 6) requires N^2 complex multiplications and $N(N - 1)$ complex additions. If we suppose that computing time is proportional to the number of multiplications, then the approximate ratio of direct to FFT computing time is given by

$$\frac{N^2}{N\frac{p}{2}} = \frac{2N}{p} \tag{15}$$

which for $N = 1024 = 2^{10}$ is a computational reduction of more than 200 to 1.

The matrix factoring procedure introduces one discrepancy. The computation of Equation 9 yielded $\overline{H(k)}$ instead of $H(k)$; that is,

$$\overline{H(k)} = \begin{bmatrix} H(0) \\ H(2) \\ H(1) \\ H(3) \end{bmatrix} \text{ instead of } H(k) = \begin{bmatrix} H(0) \\ H(1) \\ H(2) \\ H(3) \end{bmatrix} \tag{16}$$

This rearrangement is inherent in the matrix factoring process, but unscrambling $\overline{H(k)}$ to obtain $H(k)$ is straightforward. Rewrite $\overline{H(k)}$ by replacing argument (k) with its binary equivalent,

$$\begin{bmatrix} H(0) \\ H(2) \\ H(1) \\ H(3) \end{bmatrix} \text{ becomes } \begin{bmatrix} H(00) \\ H(10) \\ H(01) \\ H(11) \end{bmatrix} \tag{17}$$

Observe that if the binary arguments of Equation 17 are flipped or bit reversed (i.e., 01 becomes 10, etc.), then

$$\overline{H(k)} = \begin{bmatrix} H(00) \\ H(10) \\ H(01) \\ H(11) \end{bmatrix} \text{ flips to } \begin{bmatrix} H(00) \\ H(01) \\ H(10) \\ H(11) \end{bmatrix} = H(k) \tag{18}$$

For N greater than 4, it can become cumbersome to describe the matrix factorization process in equation form. For this reason we often interpret Equation 9 in graphical manner. Reference 3 describes this process in detail.

Decimation in time (data enters in bit order reversed form, processes through a series of multiplications and additions, and emerges in a correctly ordered spectrum sequence) is one of many forms of the FFT algorithm. As data flows through the processor, the designer of the FFT system is faced with a significant choice. At each intermediate stage the results of the succeeding operation can be stored in a memory dedicated to that purpose, or, the results of an intermediate operation can be placed back in the same memory from which the original data came. The latter case is the more efficient solution and, although certain gains in speed can be attained by providing intermediate storage, many processing systems choose to use "in-place computation" as an optimum tradeoff between speed and cost.

One of the many forms of the FFT treats data in terms of complex pairs of points with the final unscrambling giving the desired results of either one or two data input tranforms. Real data inputs can be treated in complex fashion by loading, for example, alternate samples to their respective arrays. Some gain in speed efficiency can be achieved by treating data in complex pairs.

SMOOTHING THE FFT

In the calculation of the FFT, a finite block of data (memory period) is typically processed in block fashion, whether it be done by computer software or using special-purpose hardware. Transform averaging, if performed, may be on either a non-redundant or overlap basis; but in either case, blocks of data are processed.

The size of the block of data being processed determines the frequency resolution achieved while performing the FFT process. A signal which completes exactly one period within the data block determines the lowest frequency which can be analyzed and also the maximum resolution which can be achieved. If aliasing terms are neglected, the number of computed Fourier coefficients when operating on real, sampled data points is exactly one-half the number of sample points in the block. Using 1024 real points, the maximum number of spectrum lines which can be achieved is 512. If the 1024 data points are collected at a sample rate of 2048 samples/second, it will take exactly one-half second to sample 1024 points, effectively giving a memory period of one-half second. The lowest frequency, under these conditions, which could be processed is 2 Hz which is also the maximum resolution which could be achieved. For this example then, an analysis can be performed up to a maximum frequency of 1024 Hz with 512 frequency resolution elements (Δf) each spaced precisely 2 Hz apart.

The Fourier transform of a periodic waveform will have a spectrum component in only one cell if the periodic waveform loaded into the memory block is perfectly periodic within this block window. Spectrum smearing can take place, however, as soon as a data signal's period no longer precisely matches the memory period or an integer submultiple of the memory period. If this data is not handled properly, spectrum contributions can occur strictly as a function of sampling rate and block length independent of actual data characteristics.

The resolution of the FFT is dependent on sample rate and block length, since an FFT process performs a spectrum analysis through multiplication of sampled values by sine and cosine values without use of an actual analog filter. A means must be employed to reduce spectrum broadening (smearing) caused by signal truncation to manageable proportions. A technique, called windowing or weighting, is used to control the effects of this smearing. The relative shape of the analysis filter resulting from an FFT process depends strictly upon the weighting function employed. The absolute filter bandwidth, however, is determined by a combination of sample rate, block length and the selected weighting function. Variation of effective bandwidth as a function of the periodicity of the signal being analyzed within the data block is just as important to a spectrum analysis as the side-lobe energy produced through weighting. Window consistency in spectrum analysis is very important. Table 1 compares the parameters of different time weighting windows with special emphasis given to filter characteristics and side-lobe reduction.

The finite block length of a digital memory in effect creates a rectangular window if no additional weighting is used. A rectangular function in the time domain produces a $\frac{\sin x}{x}$ function in the frequency domain. Rectangular windowing generates additional spectrum components which will include a component with a side lobe only 13 dB below the full scale actual data signal. The width of the effective filter created by this window will not be consistent because there are notches in the effective filter shape created indicating nulls in the frequency domain. However, rectangular weighting can be used in cases where external sampling is employed such that certain periodic components are automatically periodic within the window of the processor. Rectangular weighting is quite effective when every periodic component falls at a "perfect filter" location.

Many other weighting functions have been developed to minimize truncation effects, provide a consistent filter shape, and at times, also optimize some other parameter. Because it is easy to implement, the most common window used is the Hanning, which was developed by Julius von Hann. The Hanning window is defined by

$$W(n) = 0.5 + 0.5 \cos(n\, 2\, \pi/N) \quad \text{where } n = -N/2, \ldots -1, 0, 1, \ldots N/2$$

The Hanning window, in fact, reduces the side-lobe energy from that of rectangular weighting such that the highest side lobe is down 32 dB below full scale. The width of the effective filter created by this window, like the rectangular window filter shape, will not be exactly consistent because there are notches in the effective filter shape created indicating "nulls" in the frequency domain.

Note that the Hanning window is described in terms of a cosine function. No special registers are needed to implement this weighting during the FFT process, since the cosine values stored in the FFT processor can be used for weighting purposes as well as for Fourier coefficient calculation.

The Kaiser-Bessel window, developed by J. Kaiser of Bell Telephone Laboratories, which uses a modified Bessel function of the zero'th order, first kind, was recently implemented in microprogrammed hardware by Professor fred j. harris of San Diego State University. The Kaiser-Bessel window, with the important characteristic of optimizing main-lobe energy while reducing the energy in the side lobes, produces no side lobe greater than -72 dB from full scale.

The importance of window consistency in signal analysis is illustrated in the following example. Two signals, 60 dB apart in amplitude and of adjustable frequency, were summed together and processed for their spectrum content on a linear frequency, log amplitude display basis. In Figure 1 and 2, the two signals are each periodic in the window and spaced by 120 Hz or 6 cells ($6\,\Delta f$). Both the Hanning window and the Kaiser-Bessel window distinguish the presence and precise amplitudes of each component with this spacing.

The time averaged Fourier transform for the same amplitude conditions, but for the situation where the full scale component has been moved 10 Hz or one-half cell (6.5 Δf) further away from the low level signal, is shown in Figures 3 and 4. In this case, the side-lobe energy produced by the signal truncation effects has not been suppressed by the Hanning window and the low level signal has been completely masked. The Kaiser-Bessel window, on the other hand, still permits complete detailed analysis of both signal components. From this example, it is clear that the weighting function used determines the resolving power of an FFT process.

The capability of providing both rectangular and some other selectable windowing function should be a minimum FFT processor requirement. Ideally, the weighting function, to give maximum flexibility, should be selectable with different weighting characteristics available to match a specific application. This flexibility is achieved typically through interchangeable programmed read-only memories in a hardwired or microprogrammed FFT processor, or in the case of a software system, by using appropriate software subroutines.

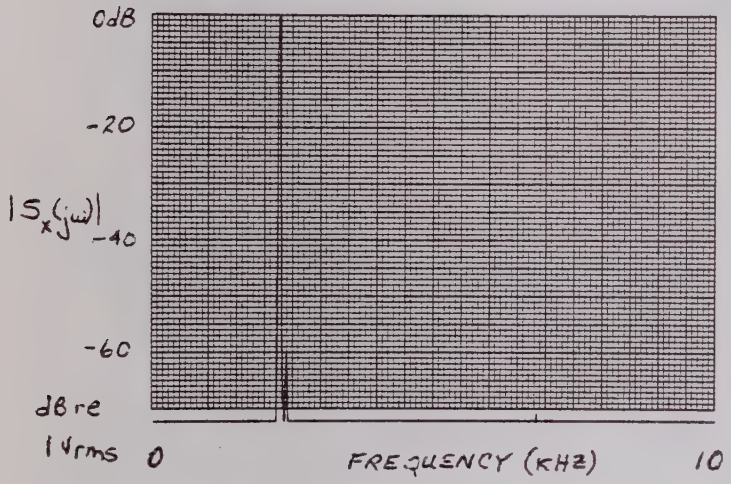


Figure 1. SPECTRUM OF TWO TONES
6 CELLS, 60dB APART
HANNING WEIGHTING

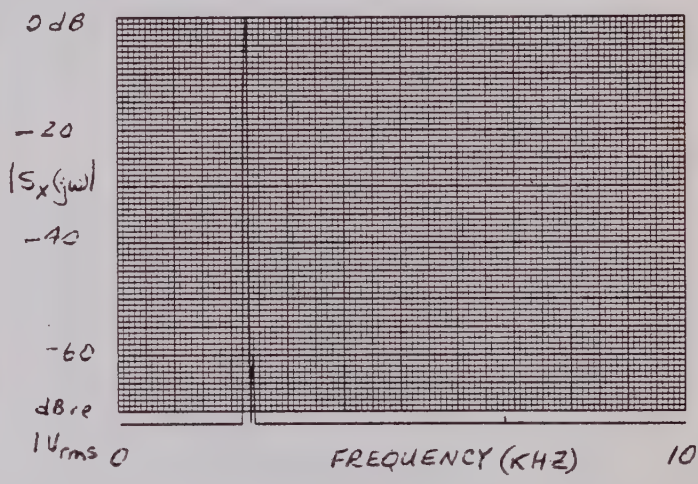


Figure 2. SPECTRUM OF TWO TONES
6 CELLS, 60dB APART
KAISER-BESSEL WEIGHTING

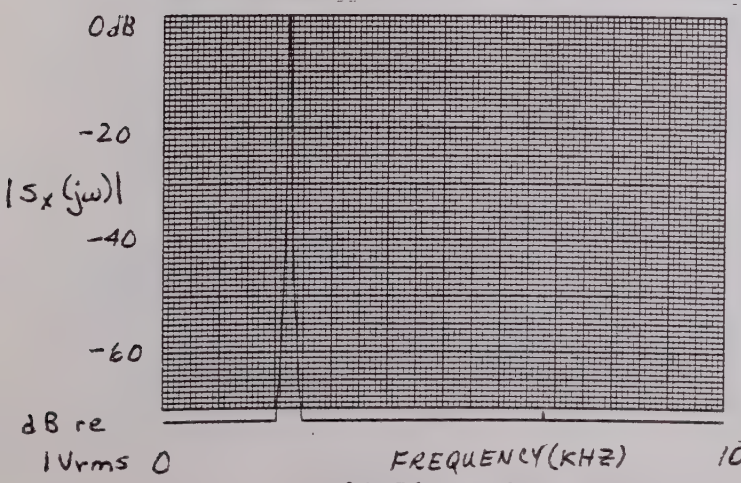


Figure 3. SPECTRUM OF TWO TONES
6½ CELLS, 60dB APART
HANNING WEIGHTING

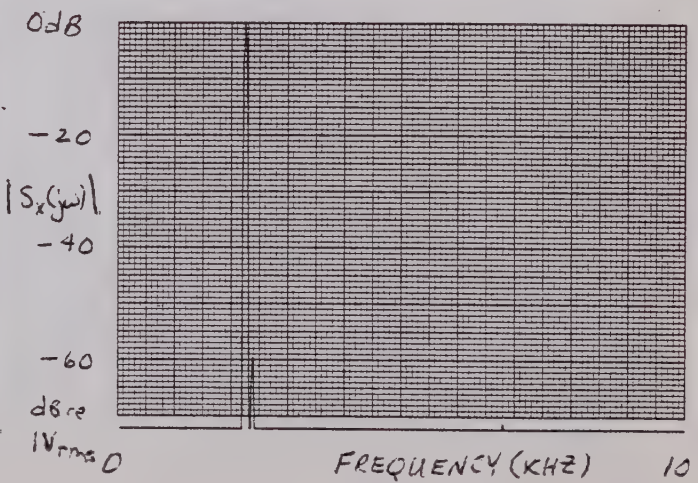


Figure 4. SPECTRUM OF TWO TONES
6½ CELLS, 60dB APART
KAISER-BESSEL WEIGHTING

SINGLE-CHANNEL APPLICATIONS

For spectrum measurements, distortion analysis, modulation, or where the existence of multiple tones is questioned, an FFT processor can be used directly to enhance single-channel measurements. An example is detection of unwanted sidebands produced by frequency modulating a carrier signal. Even though the level of the sidebands could be quite low, it is important for this application to recognize their existence, as well as the difference frequency from carrier to sideband.

The versatility and the computing power of the FFT gives it a unique capability to perform analysis of transient signals. Once a data signal has been digitally stored, preserving phase information for transient data, other manipulation and data editing can also be done prior to its processing.

A measurement of the integrated power spectrum is another convenient measurement which can be performed by an FFT processor. A question which often arises is, "Given a certain power spectrum, what is the percentage of energy in terms of the total spectrum energy which exists between two specific frequencies?" By measuring and observing the integrated power spectrum, this question can be answered quickly. A narrowband power spectrum measurement with a full scale frequency of 5 kHz is in Figure 5. The curve of Figure 6 results when this power spectrum is then integrated from low to high frequency. This curve can be calibrated to give a maximum reading in terms of total power, or very quickly be set to display 100% power maximum. The percentage of total power represented by other upper-limit frequencies can then be read directly.

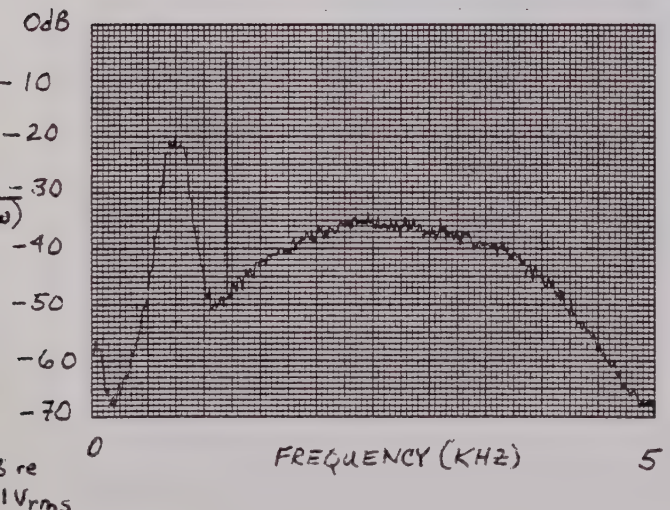


Figure 5. POWER SPECTRUM COMPOSITE SIGNAL

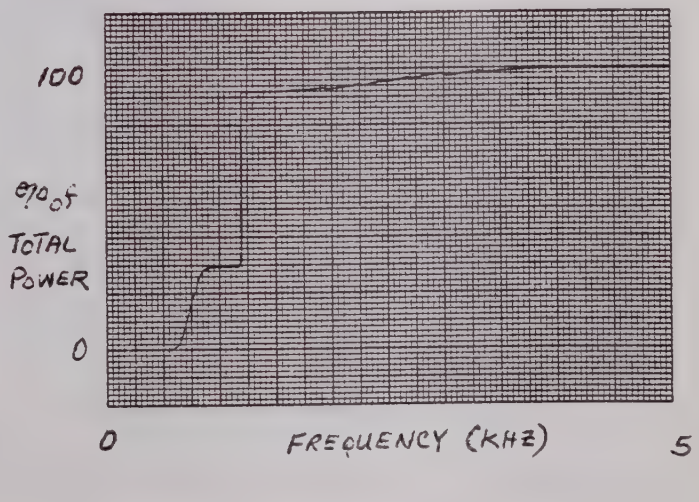


Figure 6. INTEGRATED POWER SPECTRUM

MULTI-CHANNEL APPLICATIONS

Any one of a number of phase-sensitive cross functions can be employed where mutual properties between signals are required. To determine if, in fact, a relationship does exist between two signals in the frequency domain, the cross-spectrumgives a phase and amplitude sensitive measure directly.

An example of one of the primary applications for the cross-spectrum is the use of the cross-spectrum for direct phase difference measurements. As seen in Reference 4, measurement of the cross-spectrum also serves as an inherent part of transfer function measurements. The cross-correlation function is most often used in measuring delay through a linear system. Where more than one path might possibly exist between two points of interest, cross-correlation analysis has also successfully been used to determine precise transmission paths. In this application, if more than one source is present, it is important that independent noise sources be considered, since correlation between noise sources can invalidate the interpretation of correlation levels.

The operations to be performed in a cross-correlation analysis can be thought of conceptually as delay, multiplication, and integration. The delayed signal replica is moved further and further in one direction, as longer and longer delay takes place during the integration process. However, the delay process cannot proceed indefinitely, since finite length memories are involved. For the condition of a finite memory length, memory content is rotated as the signal is processed to the end of the memory. When a signal has progressed to the end of the memory, it effectively "wraps around" and appears at the other memory end. Data which wraps around at the end of the memory suddenly appears as a true correlation coefficient in the negative delay direction when performing a correlation process. These contributions, circular correlation errors, are caused by the wrap-around phenomena. Reference 2 develops completely the theory of wrap-around error. Circular correlation errors can occur in an FFT process regardless of the exact type of implementation used.

The most straightforward means to eliminate these errors would be to eliminate any wrap-around contributing elements, which can be done by writing zeros into half of each memory involved, assuring that only zeros are contributed as wrap-around data samples. Figure 7 shows the contents of input memories after zeros have been added. The addition of zeros tends to diminish the total integrated amplitude, resulting in a tapered correlogram, since the correlation coefficient at any lag value ($\Delta \tau$) is actually the integrated cross-product of the entire data samples involved. Figure 8 is an example of a tapered cosine correlogram rather than a cosine correlogram produced by the cross-correlation between two identical sine waves.

The zeroing of a portion of memory to avoid circular correlation errors allows us to calibrate and repeat the results, but the tapering effect can cause problems in interpreting a correlogram especially in a real time processing system. For example,

when cross-correlation analysis is being used to determine principal transmission paths in a noise reduction program, the height of the cross-correlation peaks should give a direct indication of which paths should be investigated first. The peaks closer to τ_0 will almost always appear to have a larger magnitude if the taper of the correlogram is not removed. Fortunately, the unwanted taper can be removed, while still retaining the effective means of eliminating circular correlation errors, by applying a correction factor which applies no correction at the center of the correlogram and maximum correction at either end (bowtie correction). The cross-correlation function previously shown uncorrected in Figure 8 is shown in Figure 9, corrected.

The cross-correlation function, corrected and uncorrected, between two microphones located in a reverberant field including several reflections is seen in Figures 10 and 11. By correcting the taper, the interpretation of the correlation function is greatly simplified. Without the correction, an additional computation must be made before the resulting correlation peaks can be used.

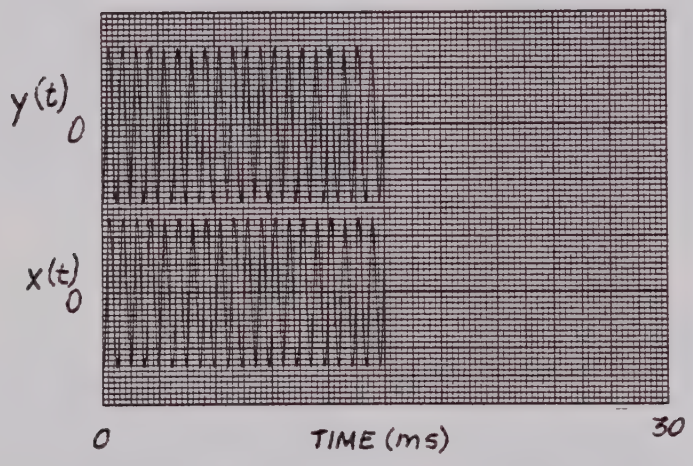


Figure 7. INPUT MEMORIES WITH ZEROS ADDED

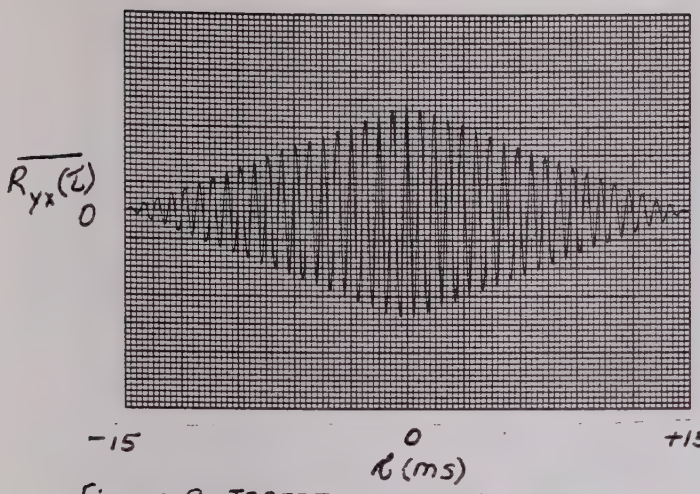


Figure 8. TAPERED CORRELOGRAM SINE INPUTS

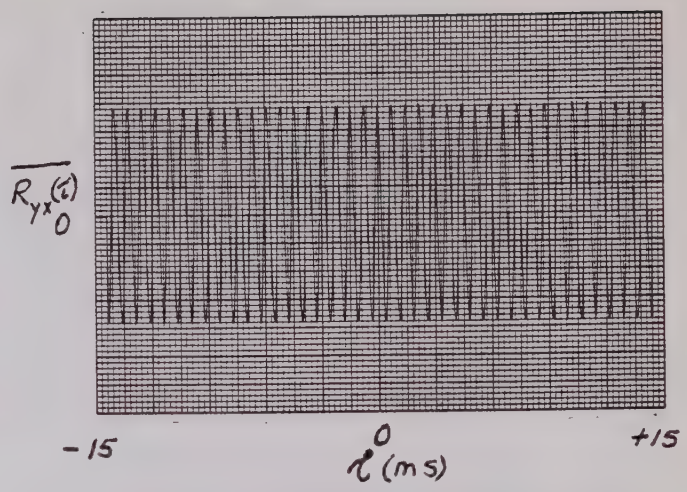


Figure 9. CORRECTED CORRELOGRAM SINE INPUTS

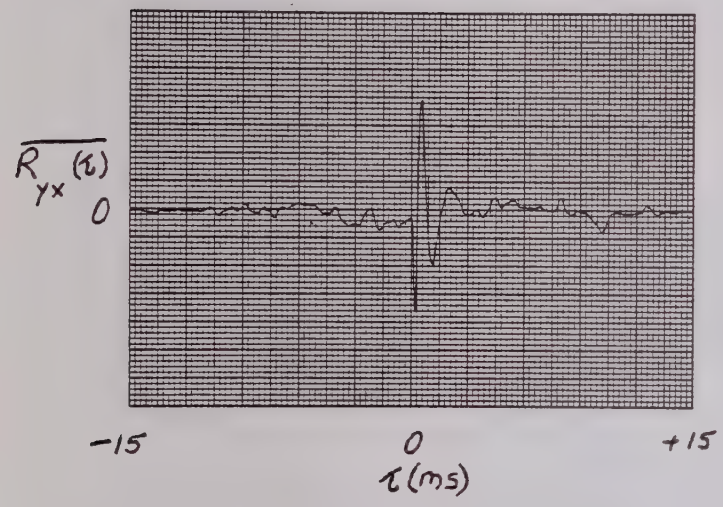


Figure 10. TAPERED CORRELOGRAM TWO PATH MEASUREMENT

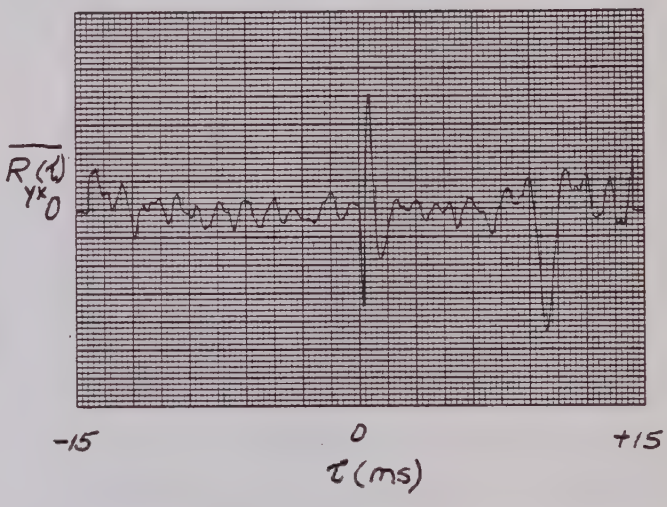


Figure 11. CORRECTED CORRELOGRAM TWO PATH MEASUREMENT

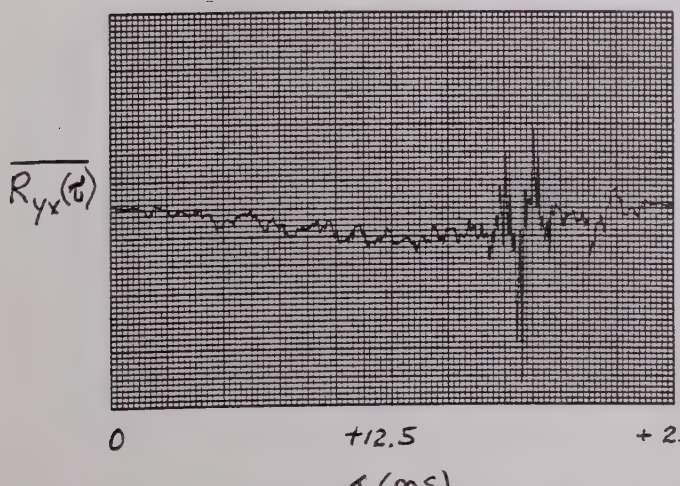


Figure 12. Expanded CROSS-CORRELOGRAM CHANNEL B DELAY

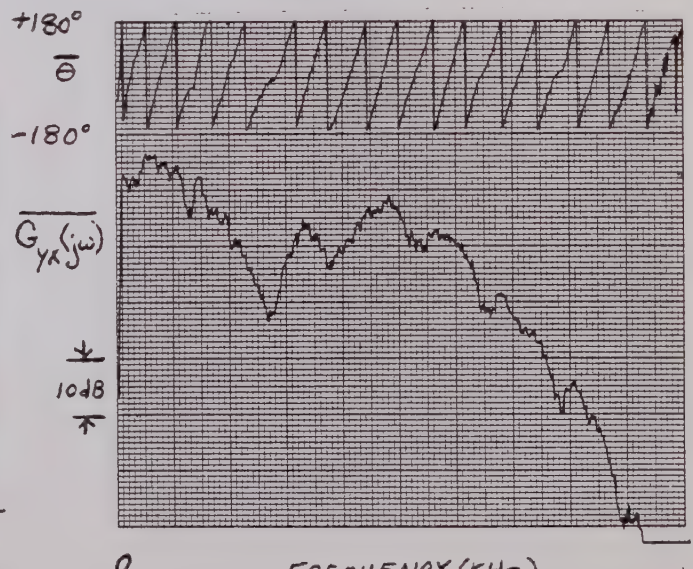


Figure 13. CROSS-SPECTRUM FUNCTION SHOWING DELAY

To ascertain that two signals are related, and if so, to begin to appreciate their relationship, the most straightforward concept is to measure the correlation coefficient. Where two different noise signals are involved, the cross-correlation coefficient gives a direct indication of any mutual properties two signals may share in the time domain. For noise signals without periodicities, the cross-correlogram gives a direct indication of delay time between signals.

For example, the sound from a single noise source was measured by two microphones approximately 20 feet apart. Since a single, broadband source was the only sound present in this case, a cross-correlogram would show strictly the time delay between signals. For additional correlogram resolution, an increased sampling rate can be used to reduce the time between samples, thus providing increased correlogram definition. Delaying one of the data channels prior to processing shifts the correlogram "window" in time such that 0 time delay is not at the center of the correlogram; therefore, the center point can represent some arbitrary time position; for an example see Figure 12. Here the total time delay between signals was greater than 12.5 milliseconds, yet a Δ t resolution of 25 microseconds was required. For these conditions, a delay of 12.5 milliseconds was introduced into the B channel signal path prior to processing and this delay has allowed a four to one expansion of the correlogram, permitting a detailed determination of the delay peak time as 18.3 ms.

Since we are most often dealing with frequency sensitive phenomena, an examination of mutual properties in the frequency domain will often be desired. Just as the cross-correlogram was a phase sensitive indicator of mutual properties in the time domain, the cross-spectrum gives an indication of joint signal characteristics as a function of frequency. Note that the cross-spectrum also does not necessarily give a causal relationship.

One of the principal applications for cross-spectrum measurements is that of accurately defining phase relationships between two signals. Earlier, the cross-correlation function between two measurement locations, spaced by approximately 20 feet, was examined for the condition of a single broadband noise source. Examining the same data in terms of its cross-spectrum gives Figure 13. Here the broadband characteristics of the noise are evident with no apparent strong periodicities. Moreover, the cross-spectrum indicates an almost continuous phase roll between channels, expectedly, since a time delay of nominally 18 milliseconds was apparent between measurement locations and a time delay in the time domain is the equivalent of a phase shift in the frequency domain.

APPENDIX D

Signal Analysis Functions Defined by the FFT

The result of performing the basic Fourier transform using the FFT process is a complex quantity regardless of whether the original data signals are considered real or complex. If we consider two possible data signals, x(t) and y(t), a complete series of analysis functions can be defined. The basic Fourier transforms for a time varying signal x(t) and a second time varying signal y(t) are

$$\mathcal{F}[x(t)] = S_x(j\omega) = a + jb \qquad S_x^*(j\omega) = a - jb \tag{1}$$

$$\mathcal{F}[y(t)] = S_y(j\omega) = c + jd \qquad S_y^*(j\omega) = c - jd$$

The basic Fourier transform for an individual signal can be examined in terms of real and imaginary parts, magnitude and phase, log magnitude and phase, or as a Nyquist display. It is important to remember that the complex phasor representation which results (Equation 19) is produced by the interaction of the single-channel data signal and the complex phasor

$$\varepsilon^{-j\omega t}$$

In general, the phase angle produced under these conditions, when dealing with only a single-channel data signal, is virtually meaningless. For two-channel measurements, this phase becomes an important measurement parameter and under certain sampling conditions single-channel phase can also take on some significance.

For many applications, especially those involving random signals, spectrum amplitudes as a function of frequency are not nearly so meaningful as is the average power contained within a given frequency band. For these instances, the power spectrum, or sometimes the power spectral density, is the desired function. For the two signals described above, the power spectrum is defined as follows

$$G_{xx}(\omega) = S_x(j\omega) \cdot S_x^*(j\omega) = (a + jb)(a - jb) \tag{2}$$

$$\overline{G_{xx}(\omega)} = \overline{a^2 + b^2},$$

$$G_{yy}(\omega) = S_y(j\omega) \cdot S_y^*(j\omega) = (c + jd)(c - jd)$$

$$\overline{G_{yy}(\omega)} = \overline{c^2 + d^2}.$$

The power spectrum is a real valued quantity which can be calculated by taking the product of a signal's Fourier transform and the complex conjugate of the same Fourier transform. Or, it can be computed by simply adding the squares of the real and imaginary parts of a signal's Fourier transform. The power spectrum is independent of phase and is usually a continuous function of frequency. Power spectral density can be thought of as the power spectrum normalized on a per unit bandwidth basis and is often presented to simplify the calculation of total power within a selected frequency interval and also to permit rapid comparison of data analyzed with different analysis bandwidths.

Where more than one signal is involved it is also important to determine any mutual relationships between signals in addition to the individual characteristics. In order to do this, a mutual-properties type of analysis must be performed. The cross-spectrum between two data signals is such an analysis and is defined by

$$G_{yx}(j\omega) = S_y(j\omega) \cdot S_x^*(j\omega) = (c+jd)(a-jb)$$

$$\overline{G_{yx}(j\omega)} = \overline{(ac+bd)} + j\overline{(ad-bc)} = \overline{CO_{yx}(j\omega)} + j\overline{QUAD_{yx}(j\omega)}$$

$$\theta_{yx}(j\omega) = \tan^{-1}\left(\frac{\overline{QUAD_{yx}(j\omega)}}{\overline{CO_{yx}(j\omega)}}\right) \quad (3)$$

The cross-spectrum is formed by taking the product of the Fourier transform of one signal and the complex conjugate of the Fourier transform of the second signal. By forming this product as a basic part of a cross-spectrum computation, a certain amount of signal-to-noise enhancement is achieved. At the same time, signals tend to be discarded which are not mutually contained in both channel inputs. While the cross-spectrum does not give a complete causality relationship between two signals, it can serve as an excellent first indication as to whether or not two signals are, in fact, related. Since the cross-spectrum is a phase-sensitive characteristic resulting in a complex product, it also serves as an excellent measurement of the relative phase between two signals. Because the cross-spectrum is computed as the product between two signals, the units of the cross-spectrum are the product of the units of each individual signal and are effectively given in terms of a squared number.

There are many cases in engineering in which we wish to detail the relationships between different points in a system, such as determining gain and phase characteristics. In this case, the transfer function is an extremely meaningful relationship and, in general, is defined as

$$H_{yx}(j\omega) = \frac{\text{Output Response}}{\text{Input Drive}} = \text{TRANSFER FUNCTION}. \quad (4)$$

Strictly speaking, the transfer function is defined in terms of the ratio of the Laplace transform of a driving function and the Laplace transform of the response function. Since we are normally interested in a system's transfer function in the frequency domain, it has become common practice to call the ratio of a system's output Fourier transform to its input Fourier transform (frequency response function), the transfer function. From a computational standpoint, the transfer function can be pictured as the ratio of the Fourier transform of an output signal to the Fourier transform of a system input signal. However, in the interest of simplified processing, this technique can be modified, giving

$$H_{yx}(j\omega) = \frac{S_y(j\omega)}{S_x(j\omega)} = \frac{S_y(j\omega) \cdot S_x^*(j\omega)}{S_x(j\omega) \cdot S_x^*(j\omega)} \qquad (5)$$

$$\overline{H_{yx}(j\omega)} = \frac{\overline{G_{yx}(j\omega)}}{\overline{G_{xx}(\omega)}} = \frac{\text{Cross-spectrum}}{\text{Input Power Spectrum}} .$$

Equation 5 points out that by rationalizing the denominator, a complex transfer function computation can be reduced to the calculation of the ratio of the cross-spectrum to the input power spectrum. This is, then, a complex quantity divided by a real quantity which gives a complex result. Note, however, that since the transfer function can be computed by dividing the cross-spectrum by the input power spectrum and the input power spectrum contained no phase information, the phase of the resulting transfer function must be identical to the phase of the original cross-spectrum. It is also interesting to note that whereas the cross-spectrum between two related random signals has a readily defined statistical variability, the corresponding transfer function measurements for a deterministic system exhibit no random fluctuations. in essence, the variability of the cross-spectrum is offset by the variability in the input power spectrum, giving a smooth, deterministic transfer function measurement. It was only within the past few years that this phenomena has been investigated on a real time basis with the result that transfer function measurements can typically be made on deterministic systems undergoing random excitation with only a very few spectrum averages.

Each of the previously described functions is a frequency-domain parameter and may be calculated by the FFT in a very straightforward manner. One additional frequency-domain parameter can be described. In the calculation of the cross-spectrum, it was noted that cross-spectrum computation essentially consists of the product of a Fourier transform and the complex conjugate of another Fourier transform.

The convolution spectrum, which is defined by

$$C_{yx}(j\omega) = S_y(j\omega) \cdot S_x(j\omega) \qquad (6)$$

$$= (c + jd)(a + jb)$$

$$\overline{C_{yx}(j\omega)} = \overline{(ac - bd)} + j\overline{(ad + bc)},$$

is the straight product of two Fourier transforms rather than the product of a Fourier transform and the complex conjugate of another Fourier transform.

The convolution spectrum has not found wide application in engineering or generalized harmonic analysis, but rather is a necessary step in the process of convolving two time functions. Recall that convolution in the time domain is the equivalent of multiplication in the frequency domain. The existence of a convolution spectrum, however, does serve to remind us that each of the functions previously described as frequency-domain parameters has an equivalent time-domain representation. The FFT algorithm performs a direct frequency-domain transformation and it is normally a straightforward matter to perform an inverse transform on a selected spectrum computation. The time-domain equivalents of these frequency-domain functions will now be discussed.

If the Fourier transform of the original signals produced the complex quantities $S_x(j\omega)$ and $S_y(j\omega)$, the transform operation can be followed by an inverse transformation (7)

$$\mathcal{F}^{-1}[S_x(j\omega)] = x(t),$$

$$\mathcal{F}^{-1}[S_y(j\omega)] = y(t).$$

In other words, if a time-domain signal is successively transformed and then followed by an inverse transform operation, the original signal is recovered. The performance of a forward and inverse transform in sequence can, in fact, be an excellent check of a Fourier transform processor. Note that rectangular weighting must be used in the forward transform process or the original time function becomes distorted after the inverse is taken.

From a frequency-domain standpoint, the next process often performed is a power spectrum computation. If a power spectrum analysis is done, followed by an inverse transform, we obtain (8)

$$\mathcal{F}^{-1}[\overline{G_{xx}(\omega)}] = \overline{R_{xx}(\tau)}.$$

How interesting! We've seen that if a signal is transformed via the Fourier transform and then an inverse transform taken, the original signal is recovered. However, if after performing the Fourier transform operation, a simple spectrum squaring takes place prior to performing an inverse transformation, the original signal is not recovered. We obtain the signal's autocorrelation function instead.

Since the power spectrum is a phaseless, real-valued quantity and the autocorrelation function can be calculated from the power spectrum, it should be obvious that the autocorrelation function is also insensitive to phase variations. In the theory of generalized harmonic analysis, the Wiener-Khintchine theorem shows that the autocorrelation function and the power spectrum are Fourier transforms one of the other. As a straight time-domain representation, the autocorrelation function can also be described as

(9)

$$R_{xx}(\tau) = \frac{1}{T_1} \int_{-T_1/2}^{T_1/2} x(t)\, x(t+\tau)\, dt \, .$$

The autocorrelation function describes the time-domain relationship between a signal and a delayed replica of the signal. It can serve as a direct indicator of signal bandwidth, the existence of periodicities in a signal and the shape of any such periodicities. In the past, autocorrelation functions were often computed by a direct delay and multiplication process. With the FFT, however, some of the analyses previously done in the correlation domain can be performed more easily using the wider dynamic range and straightforward interpretation of the power spectrum.

A cross-spectrum analysis was seen to provide mutual properties between signals in the frequency domain. We can also take the inverse transform of the cross-spectrum to give the cross-correlation function as follows.

(10)

$$\mathcal{F}^{-1}\left[\overline{G_{yx}(j\omega)}\right] = \overline{R_{yx}(\tau)} \, .$$

The cross-correlation function describes the mutual properties between two signals in the time domain. For two completely unrelated signals, the average cross-correlation function will tend to zero. Cross-correlation functions give a direct indicator of time delay through a system and are widely used in signal detection or enhancement applications. Expressed in the time domain, the cross-correlation function is

(11)

$$R_{yx}(\tau) = \frac{1}{T_1} \int_{-T_1/2}^{T_1/2} y(t)\, x(t+\tau)\, dt \, .$$

It is interesting to note that four different types of correlation function can be described depending on the application. First, if two general data signals are correlated, the result will show their mutual relationship in either a random or a periodic sense. Mutual relationships in a random sense might show up as a time delay buildup at some point other than zero time delay. If a periodic signal is transmitted, often it is desirable to recover the transmitted signal or a function of the transmitted signal free of any noise which might have been introduced through any transmission or recording media. Here is where one of the fascinating aspects of the cross-correlation function can be exploited.

Since the cross-correlation function expresses mutual properties between signals, it is sensitive only to signal components which are contained within both inputs to the defining equation. Thus, if one of the inputs to a cross-correlation function is a periodic, noise-free signal, the average cross-correlation function by definition can contain no noise! Second, for the case of a transmitted tone burst, a received echo or signal can be correlated against the original noise-free burst. If enough of the original signal is present in the returned transmission, the resulting correlogram will be the correlation function of the originally transmitted burst.

If, on the other hand, the received transmission is correlated not against the tone burst itself, but rather against the continuous wave (c-w) signal which is being gated on and off to create the tone burst, then the only possible signal which could be recovered through a cross-correlation would be the c-w component contained in the received signal. ~~On the one hand, we were able to enhance~~

(Third type of correlation function). On the one hand, we were able to enhance a burst by correlating the received signal with the original burst, but when correlating the received signal with only one component of the burst spectrum, only that component will be recovered.

For the fourth type, suppose that we wish to recover not the correlation function of the tone burst, or a specific component of the burst, but instead recover the entire burst. It would then be necessary to correlate with a signal which contains all possible harmonics of the originally transmitted tone burst. A convenient signal which contains all possible frequencies (it has a flat spectral distribution) is the delta function. It is also known that the Fourier transform of a series of impulses in the time domain is, in effect, in the frequency domain a series of impulses. If we, therefore, take a unit impulse function and repeat it at a rate exactly proportional to the tone-burst rate, every harmonic component of the tone-burst signal will be exactly recovered. This process follows.

(12)

$$\tilde{\Phi}_{xz}(\tau) = \frac{1}{T_1} \int_{-T_1/2}^{T_1/2} [o + n](t)\ \delta(t + \tau - n t_o)\, dt.$$

where $(s + n)(t)$ is a noisy signal and T_0 is the known fundamental repetition rate.

Although what has just been described is a form of cross-correlation, the resulting process has been termed signal averaging, signal enhancement or time-domain averaging. Interestingly enough, it is also an exact description of how digital sampling takes place.

Signal averaging can be a very effective processing tool since it produces only data components which are synchronous with the reference train of unit impulses. Signal averaging is often performed in an FFT processor since it is then feasible to calculate the Fourier transform of the time-averaged waveform. If a signal which contains periodic components masked by random noise is analyzed, its power spectrum will be a composite of the noise and periodic components. Even though the noise is not synchronous with the periodic waveform, it represents real power and will contribute significantly to the overall power spectrum. If, on the other hand, a signal averaging operation is performed prior to spectrum computation, the result is a "synchronous spectrum" with all noise components which are not related to the synchronous repetition rate discarded. Note that, in general, the cross-correlation function shares an additional property of the cross-spectrum; that is, it does not give a causal relationship.

The complete input/output relationship of a system in the frequency domain was defined in Equation **5** by the frequency response function. A system's input/output relationship can also be defined in the time domain. Since the frequency response function was defined as

$$H_{yx}(j\omega) = \frac{O(j\omega)}{I(j\omega)} . \qquad (13)$$

the Fourier transform of the system output is nothing more than the product of the frequency response function and the input Fourier transform. If a system is excited by a unit impulse function, an interesting response takes place. Since a perfect impulse, theoretically, contains all frequencies, it is capable of exciting the entire frequency response of a system. Again, since a perfect impulse contains all frequencies, its Fourier transform is a constant extending across the entire frequency range of interest. If, through proper scaling, the Fourier transform of this impulse is given an amplitude of unity, the following condition exists.

If $I(t) = \delta(t-0)$,

then

$$\mathcal{F}[I(t)] = \int_{-\infty}^{\infty} \delta(t-0)\, e^{-j\omega t}\, dt = 1, \tag{14}$$

and

$$O(j\omega) = H_{yx}(j\omega) \cdot 1 = H_{yx}(j\omega). \tag{15}$$

Thus for a linear system excited by a perfect impulse, the Fourier transform of the system output is equal to the frequency response function. If we now take the inverse Fourier transform of either side of Equation 15, we have a time-domain representation of how the system being analyzed would respond to a perfect impulse. This time-domain representative (the impulse response of the system) has important applications in determining the total time response of a system and is defined as

$$h(t) = \mathcal{F}^{-1}[H_{yx}(j\omega)]. \tag{16}$$

Earlier it was seen that the product of the two Fourier transforms of the two time-varying signals produced a frequency-domain function termed the convolution spectrum. The inverse transform of this convolution spectrum is the convolution function in the time domain, which is defined

$$\overline{C_{yx}(t)} = \mathcal{F}^{-1}[\overline{C_{yx}(j\omega)}]. \tag{17}$$

The convolution function is best defined in the time domain as

$$C_{yx}(t) = \frac{1}{T} \int_{-T_1/2}^{T_1/2} y(t)\, x(\tau - t)\, d\tau. \tag{18}$$

Note the striking similarity between Equations 11 and 18. Whereas cross-correlation conceptually involves three operations, time delay, multiplication and integration, convolution can be seen to add a fourth step to these operations, folding in time. Thus, after a signal is delayed in time by τ seconds (giving the delayed form $f(t + \tau)$), convolution requires that this function be folded back in time (giving $f(\tau - t)$) prior to multiplication. This extra step, folding, is significant. Whereas, the cross-correlation is an analysis of similar properties between signals, the convolution function is in one sense a prediction of a future time function of the signals.

A major application of the convolution function is, given the system's impulse response and the input forcing function, to predict the time response of a linear system. This application can be described by

$$O(t) = \frac{1}{T} \int_{-T/2}^{T/2} h(\tau)\, I(\tau - t)\, d\tau. \tag{19}$$

Although the impulse response is seen to relate to the inverse Fourier transform of a frequency response function (strictly speaking, the impulse response function is derived through the Laplace transform operation), it can also be calculated directly through the cross-correlation technique. For example, if we consider that a system has a random input signal, $f_t(t)$, and a random output signal, $f_0(t)$, we can define the cross-correlation function, $R_{io}(t)$, as (20)

$$R_{io}(\tau) = \lim_{T \to \infty} \frac{1}{2T} \int_{-T}^{T} f_i(t) f_o(t+\tau)\, dt.$$

The input/output cross-correlation theorem tells us that if we correlate a linear system input with its output, this is equivalent to convolving the unit impulse response of that system with the input autocorrelation function. This is seen as (21)

$$R_{io}(\tau) = \int_{-\infty}^{\infty} h(t)\, R_{ii}(\tau - t)\, dt,$$

where $R_{ii}(t)$ is the input autocorrelation function and $h(t)$ is the system's impulse response.

However, when examining the autocorrelation function of a broadband noise signal with a constant power spectrum, the autocorrelation function is essentially an impulse at the origin. When a random signal is delayed with respect to itself for even a very short time interval, its correlating properties diminish quickly to zero. Stated another way, if $G_{xx}(\omega) = K$, then (22)

$$R_{ii}(\tau) = 2\pi K\, \delta(\tau),$$

where $2\pi K$ is the "weight" of the delta function.

If we now substitute this impulse autocorrelation function into the input/output cross-correlation equation together with the impulse response of our system, the following condition exists. (23)

$$R_{io}(\tau) = \int_{-\infty}^{\infty} h(t)\, 2\pi K\, \delta(\tau - t)\, dt = 2\pi K\, h(\tau).$$

This equation has effectively made use of the sampling property of the Dirac delta function, which is (24)

$$\int_{a-e}^{a+e} f(t)\, \delta(t-a)\, dt = f(a).$$

Stated simply, if any function of time is multiplied by a delta function which exists at some time, a, and the product of these two functions is integrated over a time increment which includes this time, a, the result is the original function sampled at time, a.

Returning to the input/output cross-correlation equation (Equation 27), we see that the cross-correlation of a linear system which has been excited by broadband noise can be thought of as simply being the convolution of a system's impulse response with the autocorrelation of the broadband input noise. Since the broadband noise has an autocorrelation function, which is, effectively, an impulse, the result of this cross-correlation operation is the impulse response of the linear system under test. We can then use this calculated impulse response, together with an arbitrary transient forcing function input, to predict the time history of a system's output response.

There is a marked difference between the final cross-correlation and convolution functions. Although the cross-correlation function will, in general, be an asymmetrical, complex function with values both before and after zero time delay, the convolution function, on the other hand, is always zero before zero time delay. This should not be surprising since the convolution function between the unit impulse response and an arbitrary pulse input gives the predicted output of a linear system in time. We should certainly not expect an output from a system before it receives an input!

In an FFT processor, convolution operations are performed on a block basis with a calculated impulse response stored in one input memory convolved with a transient excitation pulse from a second memory.

In the calculation of the frequency response function, it was noted that the cross-spectrum which is used as part of the transfer function measurement does not give an exact causal relationship. Therefore, it is important to be sure that a transfer function measurement has been correctly performed and can be directly interpreted. In general, there are three conditions which can lead to inaccurate transfer function measurements. They are

A. The system response being measured is being caused by more than one input;

B. In addition to the measured system input and output, extraneous noise is present in the signals being analyzed; or,

C. The system under study is non-linear.

In order to protect against any of these eventualities, a qualifying function has been developed which can be used in conjunction with a measured transfer function. This qualifier, the coherence function, is defined as

$$\overline{\gamma^2_{yx}(\omega)} = \frac{|\overline{G_{yx}(j\omega)}|^2}{\overline{G_{xx}(\omega)}\,\overline{G_{yy}(\omega)}} \quad , \text{where} \quad 0 \le \overline{\gamma^2_{yx}(\omega)} \le 1. \tag{25}$$

The coherence function compares the power contained within the cross-spectrum with the power spectrum of the input and the power spectrum of the system response. If none of the three conditions noted above is present, then the coherence function will tend toward one. If any of these conditions is present, or if a multiple input/multiple output situation exists, the coherence function will diminish. In some instances, a limit below which transfer function measurements will not be accepted may be set on the coherence function. Should the coherence function fall below these limits, then a new set of measurements must be taken before the transfer function data can be used.

Whereas the coherence function is a qualifier which can be used with transfer function measurements, it cannot be used to modify measured transfer functions. There are many cases, however, where it is necessary to know the relative contribution of different incoherent sources to a total measured response. This has led to the formulation of a concept known as the Coherent Output Power spectrum. In essence, the coherent output power spectrum is a measurement of the portion of a total power spectrum which is coherent with some reference input (usually noise). The coherent output power (COP) is defined as

$$\overline{COP} = \overline{\gamma^2_{yx}(\omega)}\,\overline{G_{yy}(\omega)} = \frac{|\overline{G_{yx}(j\omega)}|^2}{\overline{G_{xx}(\omega)}} \, . \tag{26}$$

Note that the definition of COP depends on the magnitude of the output power spectrum and the magnitude of the coherence function. It does not depend on the magnitude of the reference spectrum, an extremely useful point. When several, incoherent sources are involved, it often happens that the major contributor to the measured overall response is not necessarily the reference point with the highest level. For many noise-related applications, the coherent output power is one of the most useful FFT-related parameters.

REFERENCES

(1) Brigham, E. Oran, The Fast Fourier Transform, Prentice-Hall, 1974.

(2) Enochson, Loren D. and Otres, Robert K., Programming and Analysis for Digital Time Series Data, Shock and Vibration Information Center Monograph SVM-3, 1968.

(3) harris, fred j., Trigonometric Transforms, Spectral Dynamics Corp. publication DSP005, 1975.

(4) Lee, Y.W., Statistical Theory of Communication, John Wiley & Sons, Inc., 1960.